Weekly Report

Philip Woolley

2024-10-25

Time Log Reponse:

- Investigated different method for increasing 3d registration accuracy
- What are you planning on working on next? - Continue segmenting training data. Retrain model with additional data and new metric. Change website structure to match request from Bree
- Is there anything blocking you? - Access to "Appearance" tab on wordpress needed to rearrange navigation menu

1 Abstract

Abstract

Deformable image registration (DIR) has been well explored in recent decades, and it is widely utilized in clinical tasks, especially dose warping. Nowadays, as deep learning (DL) develops rapidly, many DL-based methods were also applied in DIR. This paper reviews DL-based DIR methods in recent years and the application of DIR in dose warping. We collected and categorized the latest DL-based DIR studies. A thorough review of each category was presented, in which studies were discussed based on their supervision, advantage, and challenges. Then, we reviewed DIR-based dose warping and discussed its rationale, feasibility, successes, and difficulties. Lastly, we summarized the review on both parts and discussed their future development trend.

Summary

Citation

Haonan Xiao, Ge Ren, Jing Cai, A review on 3D deformable image registration and its application in dose warping, Radiation Medicine and Protection, Volume 1, Issue 4, 2020, Pages 171-178, ISSN 2666-5557, https://doi.org/10.1016/j.radmp.2020.11.002. (https://www.sciencedirect.com/science/article/pii/S2666555720300629)

2 Scripts and Code Blocks

3 Documentation

The VisualizeModelResults.ipynb notebook is used for creating and viewing images of model output on validation data. Users provide a pretrained model and validation dataset, and this notebook inferences all of the images in the dataset and allows the user to review the output segmentations against the ground truth manual segmentations.

The DataProcess.ipynb notebook is used for converting slicer volume files (.nrrd and .seg.nrrd) into a HuggingFace dataset for use with the pretrained Mask2Former model. Volumes should be added to the "vols" folder, and segmentation volumes should be added to the "masks" folder.

https://www.morphosource.org/projects/0000C1059?locale=enpage=11sort=publication_status_s
List of available MicroCT Datasets of anolis lizards that will be used for this project. When infrastructure for data storage is ready I will prepare documentation detailing the downloading and storage process.

https://slicermorph.github.io/ Documentation for SlicerMorph, an extension of the 3D slicer tool commonly used by Biologists. This is used for loading stacks of .tiff images as a volume in 3d slicer.

https://github.com/jmhuie/SlicerBiomech Documentation for the Dental Dynamics module, which is a 3D slicer extension for calculating tooth stress from jaw segmentations. the outputs from my segmentation pipeline will need to be compatible with this module for analysis.

4 Script Validation (Optional)

5 Results Visualization

6 Proof of Work

7 Next Week's Proposal

- Continue segmenting training data for ML panoptic segmentation model
- Develop testing script for 3D image registration for converting coordinate systems
- Reformat blog page as requested by Bree

Week 10 Document Submission

Jacob Dallaire

October 25, 2024

1. Paper

Nezami, Somayeh, Ehsan Khoramshahi, Olli Nevalainen, Ilkka Pölönen, and Eija Honkavaara. 2020. "Tree Species Classification of Drone Hyperspectral and RGB Imagery with Deep Learning Convolutional Neural Networks" *Remote Sensing* 12, no. 7: 1070. https://doi.org/10.3390/rs12071070

SUMMARY

Automating tree species classification using drone imagery, particularly through the application of 3D convolutional neural networks (3D-CNNs), has proven to be highly effective in forest science, significantly reducing the manual labor required for forest inventories. This study demonstrated that a 3D-CNN model, utilizing a combination of hyperspectral (HS) and RGB data, achieved an overall classification accuracy of 98.3% for major boreal tree species, outperforming traditional machine learning methods like multi-layer perceptrons (MLPs) in both accuracy and efficiency. The findings also indicated that while canopy height models (CHMs) were less effective in enhancing classification performance, the integration of HS and RGB data provided superior results, suggesting a potential shift in remote sensing methodologies for forest management and ecological studies.

2. Scripts

Training script for object detection model. Converts a Tensor flow model from the tensorflow model zoo trained on the COCO database to be useable with the keras interface.

```
def load_json_annotations(json_path):
    with open(json_path, 'r') as f:
        annotations = json.load(f)
    return annotations

def parse_annotations(annotation):
    bboxes = []

    # Extract image dimensions
    image_width = annotation['imageWidth']
    image_height = annotation['imageHeight']

    if annotation['shapes']:
        shape = annotation['shapes'][0]
        points = shape['points']
        # Convert points to (ymin, xmin, ymax, xmax)
        xmin, ymin = points[0]
        xmax, ymax = points[1]
```

```
            bboxes.append([ymin, xmin, ymax, xmax])  # Store as [ymin, xmin, ymax, xmax]

        # Normalize bounding box coordinates to [0, 1]
        bboxes = np.array(bboxes)
        bboxes[:, [0, 2]] /= image_height  # Normalize ymin, ymax
        bboxes[:, [1, 3]] /= image_width   # Normalize xmin, xmax

        return bboxes

    def load_image_and_labels(image_path, annotation):
        # Load the image
        image = tf.io.read_file(image_path)
        image = tf.image.decode_jpeg(image, channels=3)

        # Store the original size for later scaling
        original_size = tf.shape(image)[:2]  # Height, Width

        # Resize the image to a standard size
        image_resized = tf.image.resize(image, [320, 320])

        # Get the corresponding annotations for this image
        bboxes= parse_annotations(annotation)
        #bboxes = np.concatenate([bboxes, c_score.reshape(-1, 1)], axis=-1)
        bboxes = tf.convert_to_tensor(bboxes, dtype=tf.float32)
        return image_resized, bboxes, original_size

    def load_dataset(annotations_folder):
        bboxes = []
        images = []

        # Iterate through JSON files in the specified folder
        for filename in os.listdir(annotations_folder):
            if filename.endswith('.json'):
                json_path = os.path.join(annotations_folder, filename)
                annotation = load_json_annotations(json_path)
                img, bbox, origional_size =
    load_image_and_labels(f'F:/LizardCV/bbox/{annotation["imagePath"]}',annotation)
                bboxes.append(bbox)
                images.append(img)  # Adjust as needed for correct path

        # Create dataset from image paths and annotations
        dataset = tf.data.Dataset.from_tensor_slices((images, bboxes))

        dataset = dataset.batch(8)
        return dataset
```

```
def custom_loss(y_true, y_pred):
    # Separate bounding boxes and confidence scores
    bbox_true = tf.squeeze(y_true)[:, :4]  # First 4 values: bounding box
    conf_true = tf.squeeze(y_true)[:, 4]  # Last value: confidence score

    bbox_pred = y_pred[:, :4]  # First 4 values: predicted bounding box
    conf_pred = tf.squeeze(y_pred)[:, 4]  # Last value: predicted confidence score

    # Bounding box loss (Mean Squared Error)
    bbox_loss = tf.reduce_mean(tf.square(bbox_true - bbox_pred))

    # Confidence score loss (Binary Cross-Entropy)
    conf_loss = tf.reduce_mean(tf.keras.losses.binary_crossentropy(conf_true, conf_pred))

    # Total loss is a combination of both
    total_loss = bbox_loss #+ conf_loss
    return total_loss

# Example usage of the dataset loader
annotations_folder = 'F:/LizardCV/bbox'  # Path where JSON files are stored
train_dataset = load_dataset(annotations_folder)

# Load a pre-trained object detection model (SSD MobileNet V2 here as an example)
#base_model =
tf.keras.models.load_model('C:/Users/Dallaire/Desktop/LizardsCV/models/base.h5')
base_model = tf.keras.applications.MobileNetV2(input_shape=(320, 320, 3),
include_top=False, weights='imagenet')
# Freeze the base model layers to retain pre-trained features
base_model.trainable = False

# Add custom detection layers (you can modify the layers depending on your classes and
bounding boxes)
x = base_model.output
x = tf.keras.layers.Conv2D(256, (3, 3), activation='relu', padding='same')(x)  # Detection-
specific conv layer
x = tf.keras.layers.GlobalAveragePooling2D()(x)
x = tf.keras.layers.Dense(128, activation='relu')(x)
output_boxes = tf.keras.layers.Dense(4, activation='sigmoid')(x)  # 4 for bounding box
coords

# Create the full model for object detection
detection_model = tf.keras.Model(inputs=base_model.input, outputs=output_boxes)

# Compile the model with appropriate loss functions and an optimizer
detection_model.compile(optimizer=tf.keras.optimizers.Adam(learning_rate=0.0001 /
10), loss='mean_squared_error', metrics=['mae'])
```

```
# Train the model (initial training with frozen backbone)
history = detection_model.fit(train_dataset, epochs=10)

# Fine-tuning: Unfreeze some of the layers of the base model for further training
base_model.trainable = True
fine_tune_at = len(base_model.layers) // 2  # Unfreeze half of the layers for fine-tuning

for layer in base_model.layers[:fine_tune_at]:
    layer.trainable = False  # Keep earlier layers frozen

# Compile again with a lower learning rate for fine-tuning
detection_model.compile(optimizer=tf.keras.optimizers.Adam(learning_rate=0.0001 /
10), loss='mean_squared_error', metrics=['mae'])

# Continue training for fine-tuning
fine_tune_history = detection_model.fit(train_dataset, epochs=10)

# Save the fine-tuned model for inference
detection_model.save('F:/LizardCV/detection.h5')
```

3. Documentation

I created and object detection model that was trained with the 990 images I previously annotated with bounding boxes. Results were mixed with localization of the anoles being strong but correct bounding boxes not being generated. A few of the generated bounding boxes from testing are shown in figure 1.

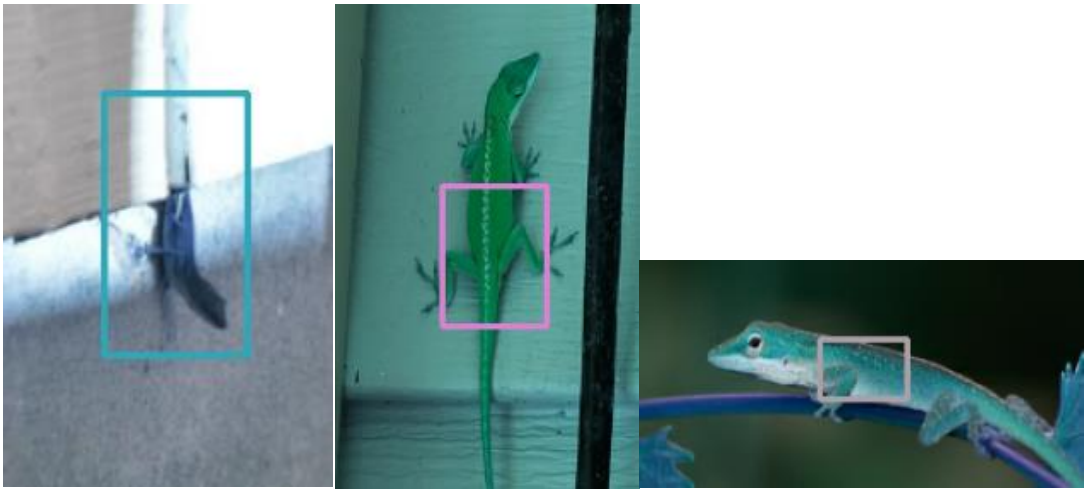


Figure 1. Predicted Bounding Boxes generated by my detection model.

The detections work best with high contrast between the anole and the background. In contexts, for example, where a green anole is on a green background the detector has much more difficulty correctly bounding the full anole.

4. Next Weeks Proposal

I will be implementing boosting for the training data by adding rotational copies for all annotations. I will also generate another ~1000 annotations. Pending success of correct bounding I will be adapting the classification model to be crop images to the area of the bounding box and boosting the classes with few samples.

Week10 report

Ruiqing Wang | CiChild CV team

- What progress did you make in the last week?
  1. Worked on DropBox folders (200G) origional videos and process current model on all videos
  2. Get all labeled videos and uploaded to DropBox
  3. Get all X, Y position from various bodyparts and visualize it
  4. rerun current model and test its performance
  5. Get data analysis on labeled videos
  6. Attend Cichild group meeting and discussed about technical details
  7. Review papers on DeepLabCut and pose estimation
  8. Help assembling paper report submissions and address submission situation.
- What are you planning on working on next?
  1. Get analysis on current labeled videos
  2. Learn new software on animal pose estimation
  3. Meet with Cichild CV team to discuss current progress
- Is anything blocking you from getting work done?

N/A

Paper abstract

Paper: SLEAP: Multi-animal pose tracking https://doi.org/10.1101/2020.08.31.276246

Abstract summary: The text discusses the development of SLEAP (Social LEAP Estimates Animal Poses), a sophisticated framework for multi-animal pose tracking that leverages deep learning techniques originally designed for computer vision. By addressing the complexities of tracking multiple interacting animals, SLEAP incorporates configurable neural network architectures and advanced inference methods, allowing for tailored performance across various experimental conditions. The framework demonstrates high accuracy in pose estimation, achieving less than 2.8 pixels of error on 95% of tracked points, and provides extensive user support, including a graphical interface and resources for training and inference.

Methodology:
Neural network model training in the SLEAP framework begins with as few as ten labeled frames, allowing users to either select from predefined configurations or customize their own, with accessible documentation on hyperparameters and output visualizations. Once trained, the model can predict animal poses on a per-frame basis, with initial predictions likely requiring corrections, which are more efficient than labeling from scratch. To solve the multi-animal problems, there are two main ways they used: top-down and bottom up:

The top-down approach in multi-animal pose estimation involves first detecting instances within a full-resolution image and cropping them to create instance-centered images, which may still contain pixels from other instances. This method uses a designated anchor body part to provide spatial context for predicting the locations of other body parts, employing a two-stage neural network framework where the first network generates multi-peak confidence maps and the second focuses on single-peak predictions for the anchored instance. While effective, this approach has limitations, including a lack of global contextual awareness and dependency on the accuracy of the initial detection stage, which can affect overall performance, particularly in complex scenes with multiple overlapping instances.

The in a bottom-up approach of Part Affinity Fields (PAFs) for multi-animal pose estimation, where PAFs represent the connectivity between body parts as a vector field. Each directed edge in the skeleton graph connects a source body part to a destination body part, and the PAFs are generated from labeled data by calculating distance-weighted edge unit vectors, which are then combined to form a comprehensive representation of body part relationships. The methodology emphasizes the importance of maintaining a spanning arborescence structure to facilitate efficient bipartite matching, thereby optimizing the assembly of body part instances while minimizing inter-node dependencies.

Scripts and Code Blocks

This week I process 200G videos in total to create all labeled walking lizards. My main job is downloading from dropbox, upload to PACE, get test videos, all postion data and move back to DropBox. Here is the script I use to process analyzed videos:

```
import deeplabcut
import os

path_config_file = "/home/hice1/rwang753/scratch/week9/trained_20videos_0603_RW_2024-10-10/20_videos_0603-RW-2024-10-10/config.yaml"
new_video_path = "/home/hice1/rwang753/scratch/week9/trained_20videos_0603_RW_2024-10-10/20_videos_0603-RW-2024-10-10/videos/05_18_2024"

#deeplabcut.analyze_videos(path_config_file,new_video_path, videotype='.MP4',save_as_csv=True)
video_files = [f for f in os.listdir(new_video_path) if f.endswith(('.MP4'))]
for video_file in video_files:
    # Construct the full path to the video file
    video_path = os.path.join(new_video_path, video_file)
    deeplabcut.create_labeled_video(path_config_file,video_path)
```

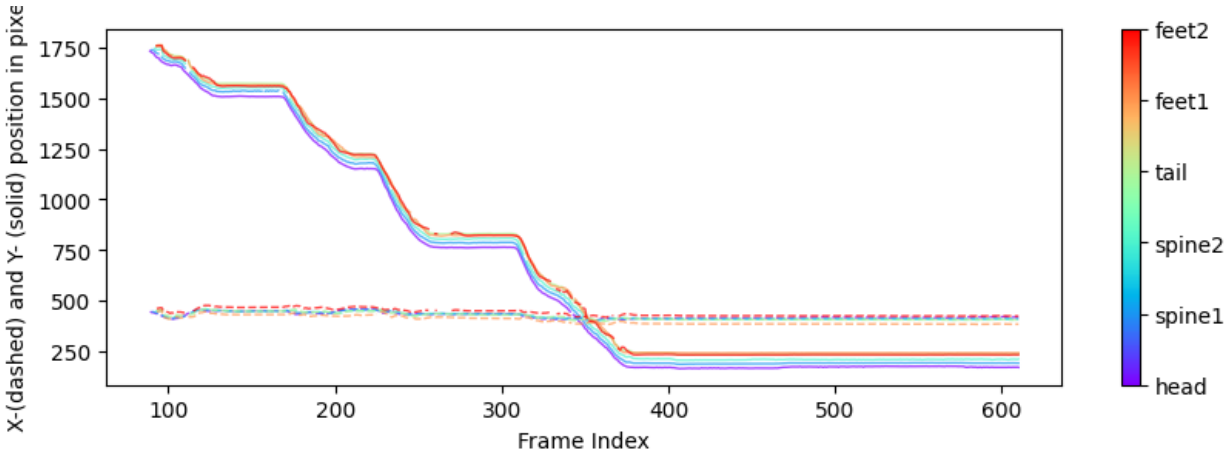
The job.sh is the same with my former code.

Documentation

The steps are pretty bald: to analyze the videos, and based on batch size and frame setting, we created labeled videos.
All my current code samples were stored in my PACE folder:
/home/hice1/rwang753/scratch/week9

Results Visualization and Code Validation

Here are some examples I got from videos analysis:



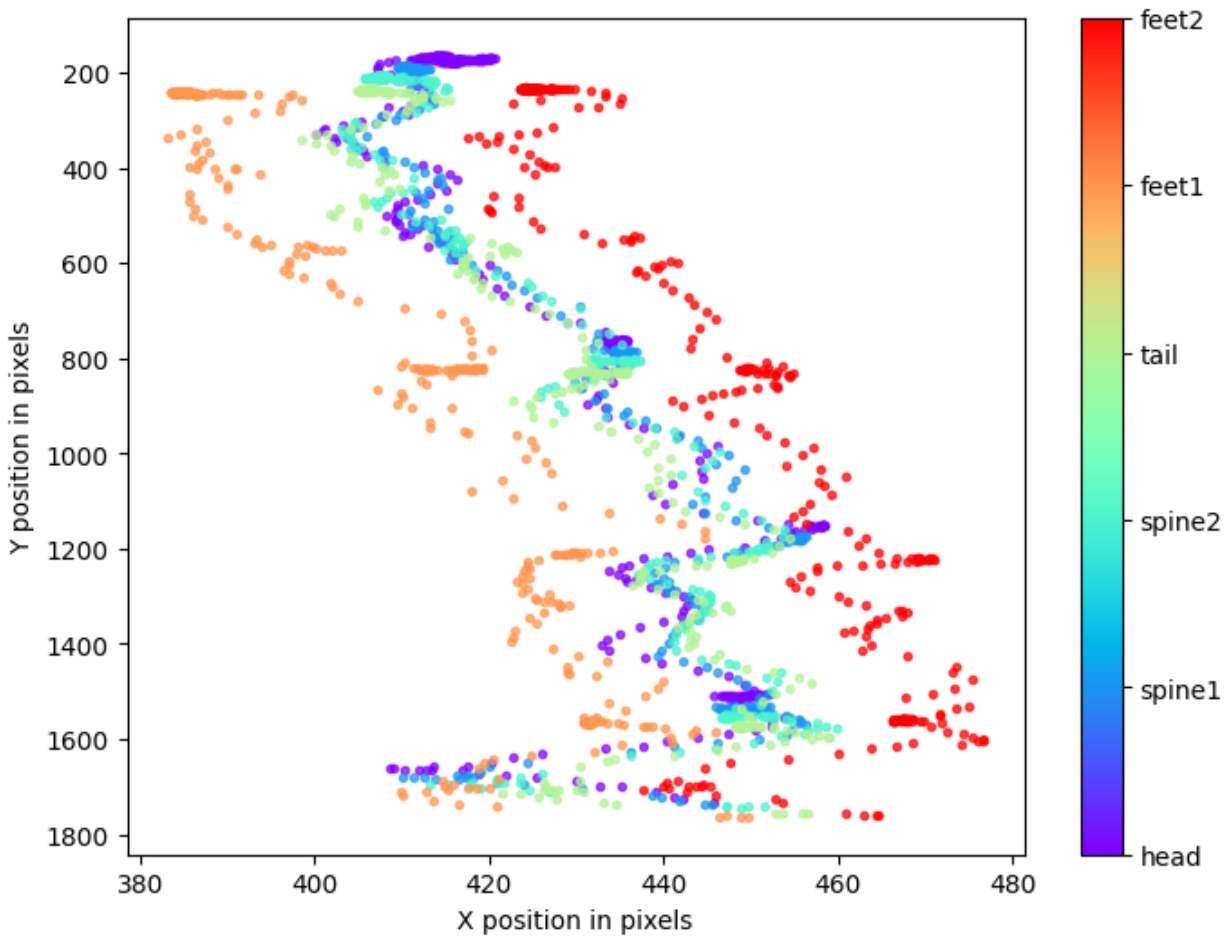


Figure 1: X-Y position likehood and X-Y in pixels of different bodyparts (video from 05_18_24: 0110_1)

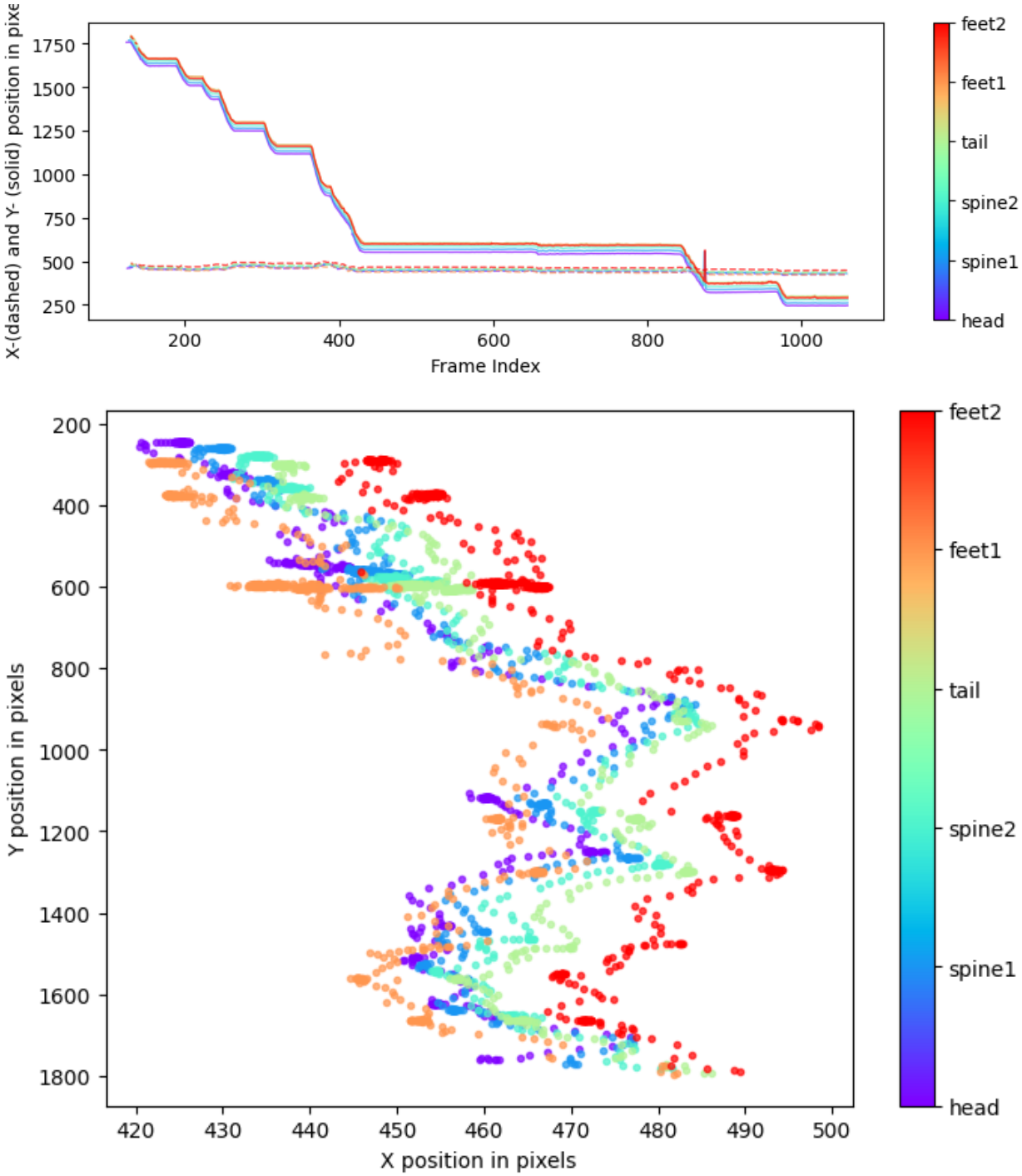
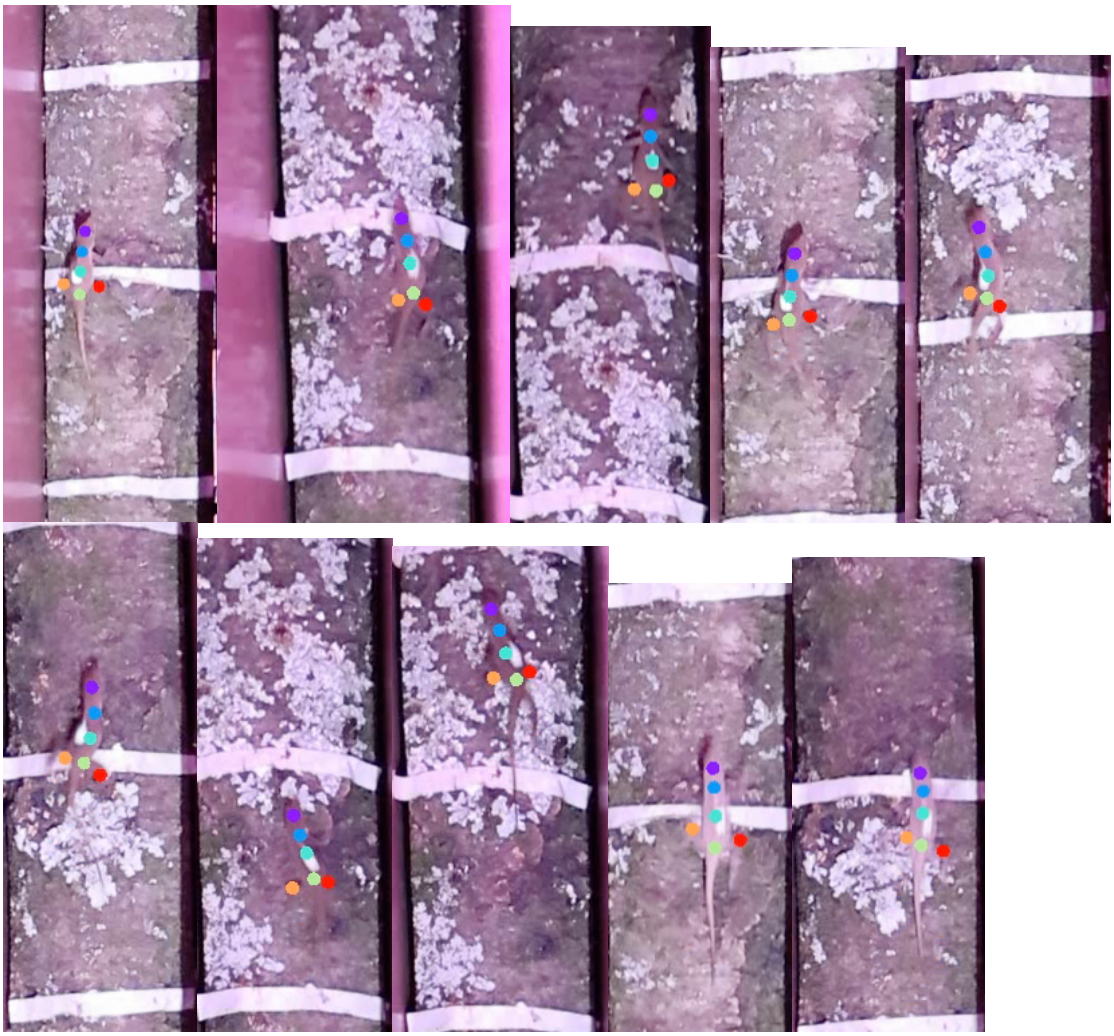


Figure 2: X-Y position likehood and X-Y in pixels of different bodyparts (video from 05_18_24: 0037_1)

Proof of Work and code validation

For the modified video process, I have modify the dot size and tested more videos, here are some screenshot:



The video I tested showed that current network is validated and could track lizard body parts in real-time movements. The screenshot above showed the current detected landmark on walking lizards.

The storage place by checking is below:

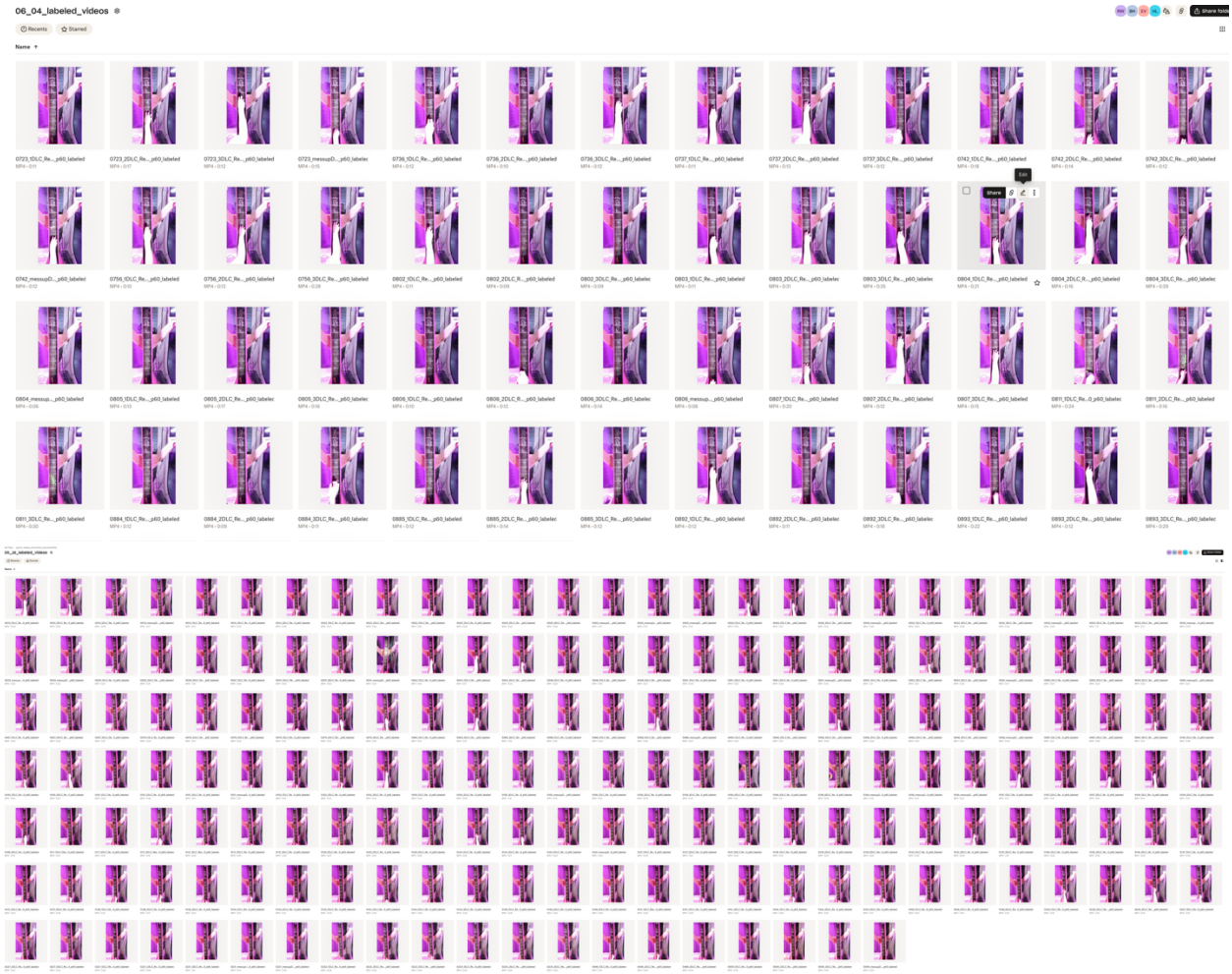
```
172.27.5.33:/ice/ice_home/9/8/pmurthy34   50T   46T  4.9T  91% /home/hice1/pmurthy34
tmpfs                                     19G     0   19G   0% /run/user/3363798
tmpfs                                     19G     0   19G   0% /run/user/3323022
172.27.5.33:/ice/ice_home/2/2/kshan9      50T   46T  4.9T  91% /home/hice1/kshan9
tmpfs                                     19G     0   19G   0% /run/user/3288402
172.27.5.33:/ice/ice_home/0/2/rwang753    50T   46T  4.9T  91% /home/hice1/rwang753
tmpfs                                     19G     0   19G   0% /run/user/3430832
172.27.5.33:/ice/ice_home/3/2/sdandekar7  50T   46T  4.9T  91% /home/hice1/sdandekar7
172.27.5.33:/ice/ice_home/9/7/jren68      50T   46T  4.9T  91% /home/hice1/jren68
```

My current work was stored at: /home/hice1/rwang753/scratch/week9

```
-rw-r--r--. 1 rwang753 gtperson    64849 Oct 21 00:25 0892_2DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.csv
-rw-r--r--. 1 rwang753 gtperson   150653 Oct 21 00:25 0892_2DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_full.pickle
-rw-r--r--. 1 rwang753 gtperson    85233 Oct 21 00:25 0892_2DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.h5
-rw-r--r--. 1 rwang753 gtperson     7029 Oct 21 00:25 0892_2DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_meta.pickle
-rw-r--r--. 1 rwang753 gtperson 16296068 Oct 21 18:31 0892_2DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_p60_labeled.mp4
-rw-r--r--. 1 rwang753 gtperson   100376 Oct 21 00:25 0892_3DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.csv
-rw-r--r--. 1 rwang753 gtperson   235406 Oct 21 00:25 0892_3DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_full.pickle
-rw-r--r--. 1 rwang753 gtperson    85233 Oct 21 00:25 0892_3DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.h5
-rw-r--r--. 1 rwang753 gtperson     7029 Oct 21 00:25 0892_3DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_meta.pickle
-rw-r--r--. 1 rwang753 gtperson 25989658 Oct 21 18:31 0892_3DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_p60_labeled.mp4
-rw-r--r--. 1 rwang753 gtperson   125095 Oct 21 00:33 0893_1DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.csv
-rw-r--r--. 1 rwang753 gtperson   289771 Oct 21 00:33 0893_1DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_full.pickle
-rw-r--r--. 1 rwang753 gtperson    85234 Oct 21 00:33 0893_1DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.h5
-rw-r--r--. 1 rwang753 gtperson     7029 Oct 21 00:33 0893_1DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_meta.pickle
-rw-r--r--. 1 rwang753 gtperson 31766269 Oct 21 18:34 0893_1DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_p60_labeled.mp4
-rw-r--r--. 1 rwang753 gtperson    71012 Oct 21 00:22 0893_2DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.csv
-rw-r--r--. 1 rwang753 gtperson   166061 Oct 21 00:22 0893_2DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_full.pickle
-rw-r--r--. 1 rwang753 gtperson    85233 Oct 21 00:22 0893_2DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.h5
-rw-r--r--. 1 rwang753 gtperson     7029 Oct 21 00:22 0893_2DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_meta.pickle
-rw-r--r--. 1 rwang753 gtperson 19734288 Oct 21 18:30 0893_2DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_p60_labeled.mp4
-rw-r--r--. 1 rwang753 gtperson   160364 Oct 21 00:28 0893_3DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.csv
-rw-r--r--. 1 rwang753 gtperson   380944 Oct 21 00:28 0893_3DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_full.pickle
-rw-r--r--. 1 rwang753 gtperson   150764 Oct 21 00:28 0893_3DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.h5
-rw-r--r--. 1 rwang753 gtperson     7029 Oct 21 00:28 0893_3DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_meta.pickle
-rw-r--r--. 1 rwang753 gtperson 39351246 Oct 21 18:32 0893_3DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_p60_labeled.mp4
-rw-r--r--. 1 rwang753 gtperson    51304 Oct 21 00:25 0893_messup2DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.csv
-rw-r--r--. 1 rwang753 gtperson   112980 Oct 21 00:25 0893_messup2DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_full.pickle
-rw-r--r--. 1 rwang753 gtperson    85232 Oct 21 00:25 0893_messup2DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.h5
-rw-r--r--. 1 rwang753 gtperson     7029 Oct 21 00:25 0893_messup2DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_meta.pickle
-rw-r--r--. 1 rwang753 gtperson 11753787 Oct 21 18:31 0893_messup2DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_p60_labeled.mp4
-rw-r--r--. 1 rwang753 gtperson    56788 Oct 21 00:32 0893_messupDLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.csv
-rw-r--r--. 1 rwang753 gtperson   125820 Oct 21 00:32 0893_messupDLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_full.pickle
-rw-r--r--. 1 rwang753 gtperson    85232 Oct 21 00:32 0893_messupDLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.h5
-rw-r--r--. 1 rwang753 gtperson     7029 Oct 21 00:32 0893_messupDLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_meta.pickle
-rw-r--r--. 1 rwang753 gtperson 14221691 Oct 21 18:33 0893_messupDLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_p60_labeled.mp4
-rw-r--r--. 1 rwang753 gtperson    62340 Oct 21 00:23 0894_1DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.csv
-rw-r--r--. 1 rwang753 gtperson   146801 Oct 21 00:23 0894_1DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_full.pickle
-rw-r--r--. 1 rwang753 gtperson    85233 Oct 21 00:23 0894_1DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.h5
-rw-r--r--. 1 rwang753 gtperson     7029 Oct 21 00:23 0894_1DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_meta.pickle
-rw-r--r--. 1 rwang753 gtperson 15561588 Oct 21 18:30 0894_1DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_p60_labeled.mp4
-rw-r--r--. 1 rwang753 gtperson    59996 Oct 21 00:42 0894_2DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.csv
-rw-r--r--. 1 rwang753 gtperson   139097 Oct 21 00:42 0894_2DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_full.pickle
-rw-r--r--. 1 rwang753 gtperson    85233 Oct 21 00:42 0894_2DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.h5
-rw-r--r--. 1 rwang753 gtperson     7029 Oct 21 00:42 0894_2DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_meta.pickle
-rw-r--r--. 1 rwang753 gtperson 15213079 Oct 21 18:37 0894_2DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_p60_labeled.mp4
-rw-r--r--. 1 rwang753 gtperson    61265 Oct 21 00:39 0894_3DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.csv
-rw-r--r--. 1 rwang753 gtperson   142093 Oct 21 00:39 0894_3DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_full.pickle
-rw-r--r--. 1 rwang753 gtperson    85233 Oct 21 00:39 0894_3DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.h5
-rw-r--r--. 1 rwang753 gtperson     7029 Oct 21 00:39 0894_3DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_meta.pickle
-rw-r--r--. 1 rwang753 gtperson 14941135 Oct 21 18:36 0894_3DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_p60_labeled.mp4
-rw-r--r--. 1 rwang753 gtperson    61394 Oct 21 00:40 test0803_1DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.csv
-rw-r--r--. 1 rwang753 gtperson   142949 Oct 21 00:40 test0803_1DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_full.pickle
-rw-r--r--. 1 rwang753 gtperson    85233 Oct 21 00:40 test0803_1DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.h5
-rw-r--r--. 1 rwang753 gtperson     7029 Oct 21 00:40 test0803_1DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_meta.pickle
-rw-r--r--. 1 rwang753 gtperson 17982221 Oct 21 18:37 test0803_1DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_p60_labeled.mp4
```

Above and below is the screenshot of some files executed in PACE.

```
-rw-r--r--.   1 rwang753 gtperson  84K Oct 22 19:22  0232_3DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.h5
-rw-r--r--.   1 rwang753 gtperson 252K Oct 22 19:22  0232_3DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_full.pickle
-rw-r--r--.   1 rwang753 gtperson 109K Oct 22 19:22  0232_3DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.csv
-rw-r--r--.   1 rwang753 gtperson 6.9K Oct 22 19:23  0144_3DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_meta.pickle
-rw-r--r--.   1 rwang753 gtperson  84K Oct 22 19:23  0144_3DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.h5
-rw-r--r--.   1 rwang753 gtperson 333K Oct 22 19:23  0144_3DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_full.pickle
-rw-r--r--.   1 rwang753 gtperson 144K Oct 22 19:23  0144_3DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.csv
-rw-r--r--.   1 rwang753 gtperson 6.9K Oct 22 19:23  0106_2DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_meta.pickle
-rw-r--r--.   1 rwang753 gtperson  84K Oct 22 19:23  0106_2DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.h5
-rw-r--r--.   1 rwang753 gtperson 300K Oct 22 19:23  0106_2DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_full.pickle
-rw-r--r--.   1 rwang753 gtperson 130K Oct 22 19:23  0106_2DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.csv
-rw-r--r--.   1 rwang753 gtperson 6.9K Oct 22 19:25  0106_5DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_meta.pickle
-rw-r--r--.   1 rwang753 gtperson 212K Oct 22 19:25  0106_5DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.h5
-rw-r--r--.   1 rwang753 gtperson 910K Oct 22 19:25  0106_5DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_full.pickle
-rw-r--r--.   1 rwang753 gtperson 386K Oct 22 19:25  0106_5DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.csv
-rw-r--r--.   1 rwang753 gtperson 6.9K Oct 22 19:26  0106_6DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_meta.pickle
-rw-r--r--.   1 rwang753 gtperson 148K Oct 22 19:26  0106_6DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.h5
-rw-r--r--.   1 rwang753 gtperson 460K Oct 22 19:26  0106_6DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_full.pickle
-rw-r--r--.   1 rwang753 gtperson 195K Oct 22 19:26  0106_6DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.csv
-rw-r--r--.   1 rwang753 gtperson 6.9K Oct 22 19:26  0500_messupDLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_meta.pickle
-rw-r--r--.   1 rwang753 gtperson  84K Oct 22 19:26  0500_messupDLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.h5
-rw-r--r--.   1 rwang753 gtperson 289K Oct 22 19:26  0500_messupDLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_full.pickle
-rw-r--r--.   1 rwang753 gtperson 127K Oct 22 19:26  0500_messupDLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.csv
-rw-r--r--.   1 rwang753 gtperson 6.9K Oct 22 19:27  0070_1DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_meta.pickle
-rw-r--r--.   1 rwang753 gtperson 148K Oct 22 19:27  0070_1DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.h5
-rw-r--r--.   1 rwang753 gtperson 429K Oct 22 19:27  0070_1DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_full.pickle
-rw-r--r--.   1 rwang753 gtperson 182K Oct 22 19:27  0070_1DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.csv
-rw-r--r--.   1 rwang753 gtperson 6.9K Oct 22 19:27  0107_2DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_meta.pickle
-rw-r--r--.   1 rwang753 gtperson  84K Oct 22 19:27  0107_2DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.h5
-rw-r--r--.   1 rwang753 gtperson 206K Oct 22 19:27  0107_2DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_full.pickle
-rw-r--r--.   1 rwang753 gtperson  89K Oct 22 19:27  0107_2DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.csv
-rw-r--r--.   1 rwang753 gtperson 6.9K Oct 22 19:28  0112_1DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_meta.pickle
-rw-r--r--.   1 rwang753 gtperson  84K Oct 22 19:28  0112_1DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.h5
-rw-r--r--.   1 rwang753 gtperson 286K Oct 22 19:28  0112_1DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_full.pickle
-rw-r--r--.   1 rwang753 gtperson 124K Oct 22 19:28  0112_1DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.csv
-rw-r--r--.   1 rwang753 gtperson 6.9K Oct 22 19:29  0035_3DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_meta.pickle
-rw-r--r--.   1 rwang753 gtperson 212K Oct 22 19:29  0035_3DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.h5
-rw-r--r--.   1 rwang753 gtperson 791K Oct 22 19:29  0035_3DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_full.pickle
-rw-r--r--.   1 rwang753 gtperson 339K Oct 22 19:29  0035_3DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.csv
-rw-r--r--.   1 rwang753 gtperson 6.9K Oct 22 19:29  0226_1DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_meta.pickle
-rw-r--r--.   1 rwang753 gtperson  84K Oct 22 19:29  0226_1DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.h5
-rw-r--r--.   1 rwang753 gtperson 211K Oct 22 19:29  0226_1DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_full.pickle
-rw-r--r--.   1 rwang753 gtperson  92K Oct 22 19:29  0226_1DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.csv
-rw-r--r--.   1 rwang753 gtperson 6.9K Oct 22 19:30  0035_1DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_meta.pickle
-rw-r--r--.   1 rwang753 gtperson  84K Oct 22 19:30  0035_1DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.h5
-rw-r--r--.   1 rwang753 gtperson 273K Oct 22 19:30  0035_1DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_full.pickle
-rw-r--r--.   1 rwang753 gtperson 118K Oct 22 19:30  0035_1DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.csv
-rw-r--r--.   1 rwang753 gtperson 6.9K Oct 22 19:30  0055_2DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_meta.pickle
-rw-r--r--.   1 rwang753 gtperson  84K Oct 22 19:30  0055_2DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.h5
-rw-r--r--.   1 rwang753 gtperson 306K Oct 22 19:30  0055_2DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_full.pickle
-rw-r--r--.   1 rwang753 gtperson 132K Oct 22 19:30  0055_2DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.csv
-rw-r--r--.   1 rwang753 gtperson 6.9K Oct 22 19:31  0026_3DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_meta.pickle
-rw-r--r--.   1 rwang753 gtperson  84K Oct 22 19:31  0026_3DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.h5
-rw-r--r--.   1 rwang753 gtperson 204K Oct 22 19:31  0026_3DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200_full.pickle
-rw-r--r--.   1 rwang753 gtperson  89K Oct 22 19:31  0026_3DLC_Resnet50_20_videos_0603Oct10shuffle1_snapshot_200.csv
drwxr-xr-x. 184 rwang753 gtperson  12K Oct 22 20:19  05_18_plot_poses
drwxr-xr-x.   2 rwang753 gtperson  32K Oct_23 00:37  05_18_labeled_videos
```



Above is one of the labeled video folders I uploaded to DropBox.

Next Week's Proposal

1. Get analysis on current labeled videos
2. Learn new software on animal pose estimation
3. Meet with Cichild CV team to discuss current progress

Week 10 Document Submission

Lizard X-RAY Landmark Group

Mercedes Quintana

What progress did you make in the last week?

- Continued to work on website
- Trained models on both image sets
- Troubleshooted issues with output from models

What are you planning on working on next?

- Implement solution for strange model behavior
- Train models again
- Continue to update the website

Is anything blocking you from getting work done?

- Nope

Abstracts:

URL: https://www.jmlr.org/papers/volume24/20-1355/20-1355.pdf

AutoKeras: An AutoML Library for Deep Learning

To use deep learning, one needs to be familiar with various software tools like TensorFlow or Keras, as well as various model architecture and optimization best practices. Despite recent progress in software usability, deep learning remains a highly specialized occupation. To enable people with limited machine learning and programming experience to adopt deep learning, we developed AutoKeras, an Automated Machine Learning (AutoML) library that automates the process of model selection and hyperparameter tuning. AutoKeras encapsulates the complex process of building and training deep neural networks into a very simple and accessible interface, which enables novice users to solve standard machine learning problems with a few lines of code. Designed with practical applications in mind, AutoKeras is built on top of Keras and TensorFlow, and all AutoKeras-created models can be easily exported and deployed with the help of the TensorFlow ecosystem tooling.

Summary: This paper introduces a tool that uses a gui system to make deep learning more accessible to a novice user. They have also implemented an algorithm to help the user find the best hyperparameters for their model.

Scripts and Code Blocks:

I only reused and visualized aspects of code already available on the github this week. When Dr. Porto revamped the ml-morph learning library, I was not aware of some of the changes made. I

have figured out the problem, which is that the bounding boxes that used to be part of the pipeline were causing strange behavior, and, with the help of Dr. Porto, will fix the issue and train the models again with better results.

Documentation:

I have no new code to document as of now.

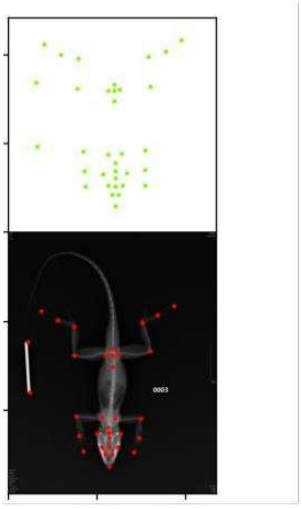
Script Validation:

I have no validation steps now.

Results Visualization / Proof of Work:

Here are some examples shared in the bi-weekly group meeting of the strange results I was getting from the output of the model. I am sure that the behavior is from how the files are originally prepared to be trained, and am in contact with Dr. Porto who pledged to help.

I trained the models again and still got unexpected results



Next Week Proposal:

I plan to keep working on the website to keep it updated with the new meetings and work done. I will train both models so I will be able to decide which image set is better to keep.