Week 4 Report

Wen Han Chia
(Lizard Classification)

Time-Log

Additionally, the time-log should include any work you've done for your *role* work (e.g., meeting management, web management, programs management, etc.)

What did you do this week?

- Meeting Manager Role
  - Coordinated with Higher Ed team regarding team channel and sent previous meeting recording to Higher Ed team
  - Joined HAAG Youtube channel – will upload meetings to the channel in the future
- Preliminary Data analysis and annotation
  - Attained the count of each classes
  - Annotated bounding boxes on dataset using Roboflow
    - Experimented with Auto-labelling feature on Roboflow
    - Implemented a workflow to upload and annotate dataset using Roboflow
- Explored PACE ICE access that was only granted this week

What are you going to do next week

- Continue annotating lizard dataset
- Will explore fine-tuning vision models using PACE ICE

Blockers, things you want to flag, problems, etc.

Abstracts:

Singh, Bharat, et al. *R-FCN-3000 at 30fps: Decoupling Detection and Classification*. arXiv:1712.01802, arXiv, 5 Dec. 2017. *arXiv.org*, https://doi.org/10.48550/arXiv.1712.01802.

We present R-FCN-3000, a large-scale real-time object detector in which objectness detection and classification are decoupled. To obtain the detection score for an RoI, we multiply the objectness score with the fine-grained classification score. Our approach is a modification of the R-FCN architecture in which position-sensitive filters are shared across different object classes for performing localization. For fine-grained classification, these position-sensitive filters are not needed. R-FCN-3000 obtains an mAP of 34.9% on the ImageNet detection dataset and outperforms YOLO-9000 by 18% while processing 30 images per second. We also show that the objectness learned by R-FCN-3000 generalizes to novel classes and the performance increases with the number of training object classes - supporting the hypothesis that it is possible to learn a universal objectness detector. Code will be made available.

What did you do and prove it

- Explored auto-labelling feature on Roboflow
    - First the images of each classes of Anole had to be separated into folders before they were uploaded to Roboflow
    - Next, I had to provide a prompt and adjust the confidence level of the Zero-shot model that is responsible for labelling this dataset
    - Result:
        - Managed to label 1000 images of the Knight Anole classes using the free credit (Image 1)
        - A quick visual analysis shows that the annotated bounding box were largely accurate. The auto-labelling fails when the lizard is hard to be seen, even by a human inspection too(e.g. Image 2).
        - If a model fails, I will label the species manually (e.g. Image 3).
        - I am still in the midst of verifying these dataset

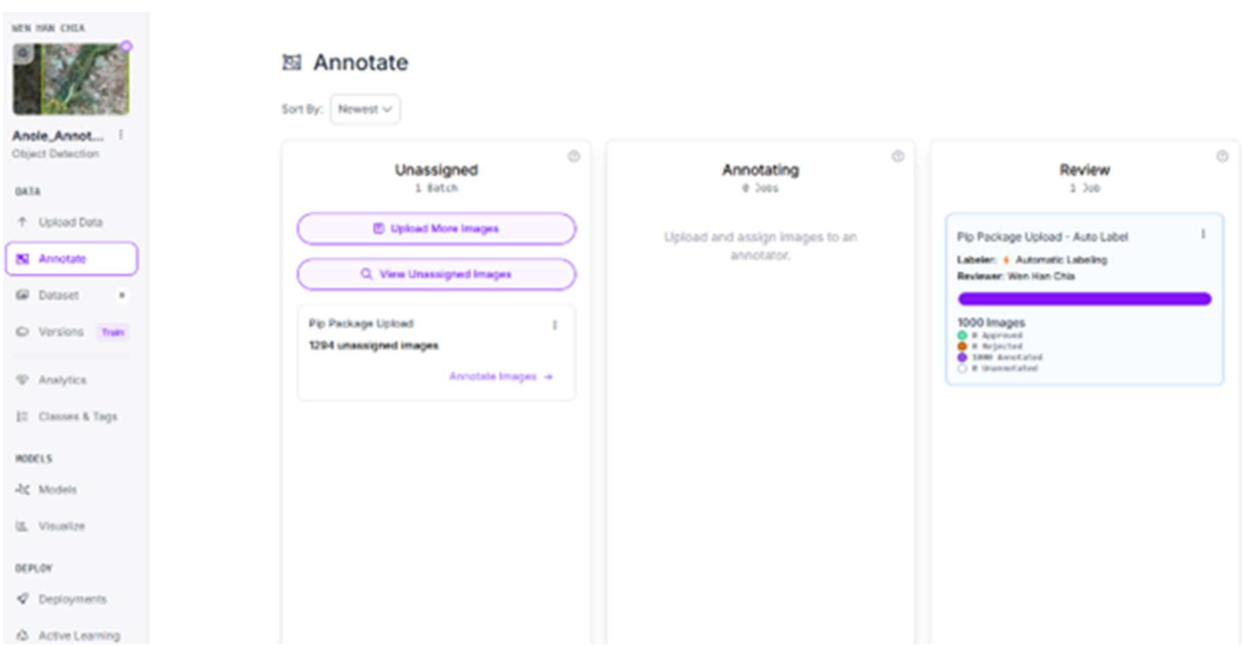

Image 1

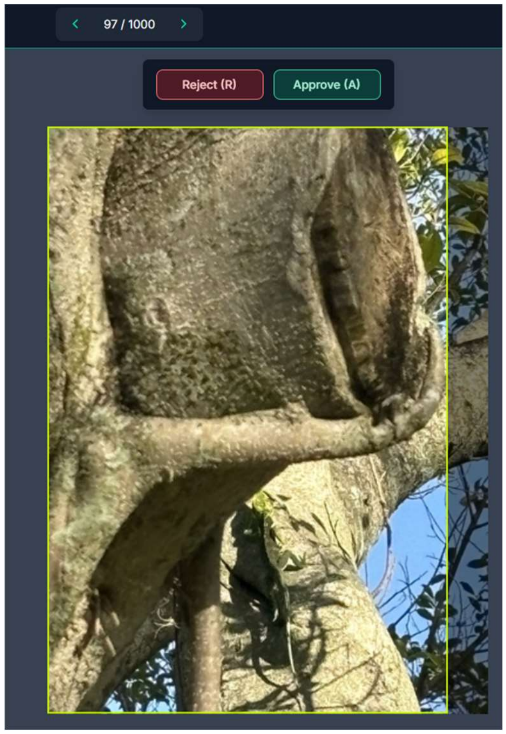

Image 2

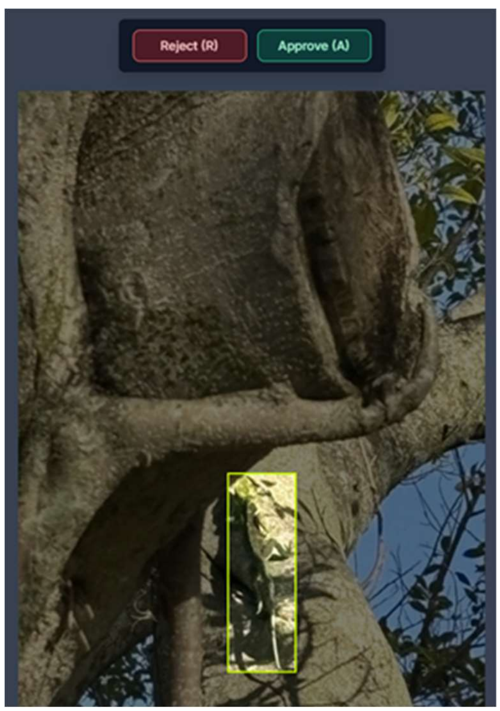

Image 3