Minjae Cho

Mr. Ray

ISM

17 April 2023

Hardhat Detection

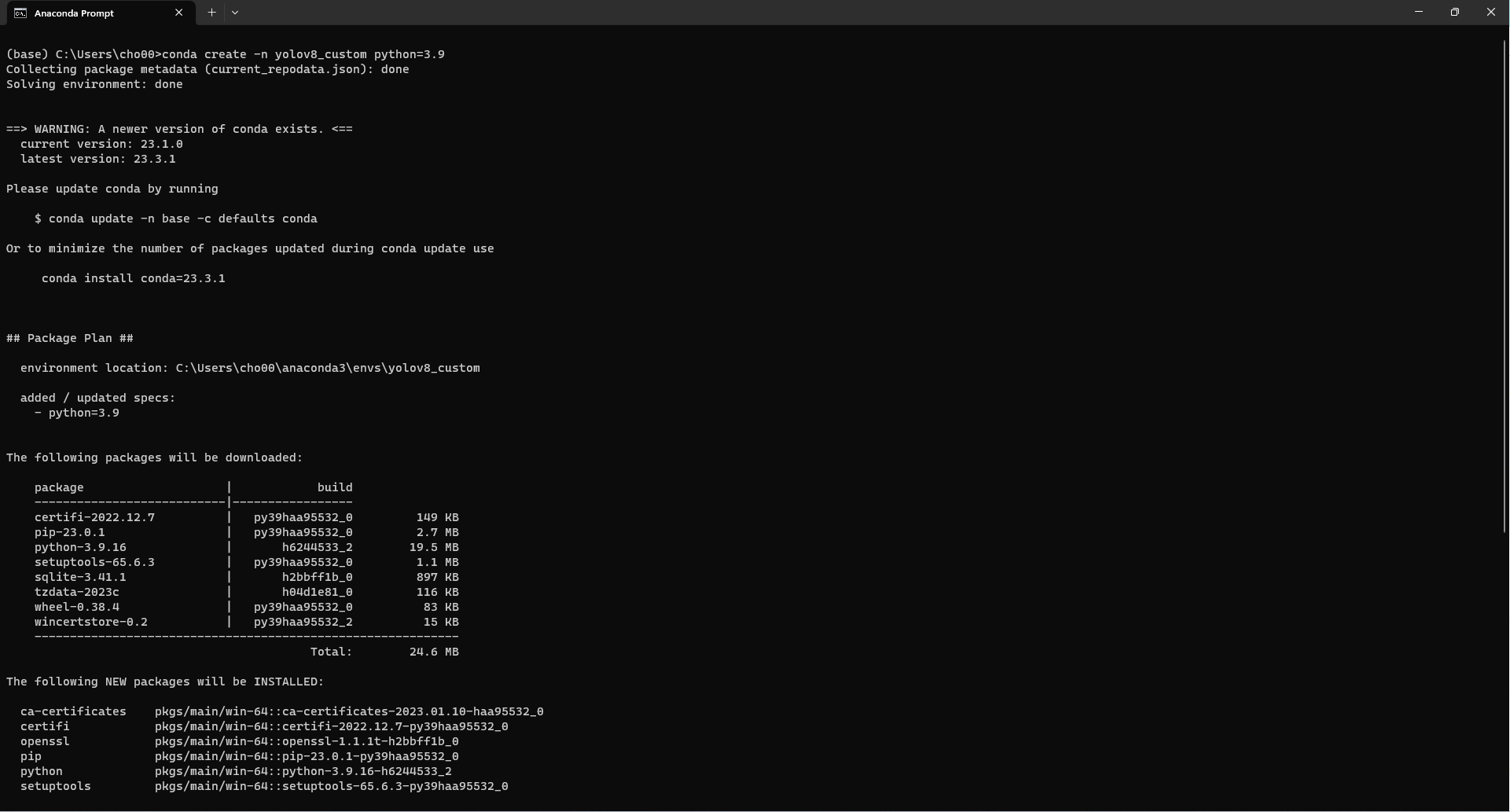
# Getting Started

While creating the product during the first semester of ISM, new information was found through research and interviews, such as the fact that there are many cameras on construction sites that overlook workers all the time. With that information, Original Work could be improved significantly on its practical application, as the Original Work did not consider practical applications. By creating an accessible mobile application that utilizes hardhat detection, more people will have access to it, although it would be best utilized by Safety Officers who monitor construction site workers. Rather than focusing on the detection of general objects of construction sites, hardhats will be the focus as it is one of the most crucial PPE and a lot of workers overlook wearing them.

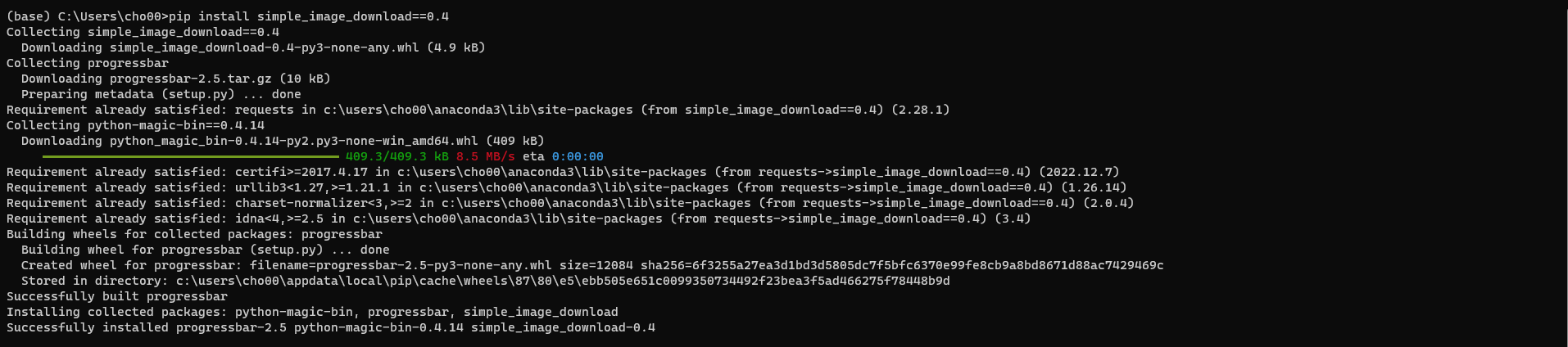
Similar to the Original Work, the YOLO algorithm will be used in this project. Since then, however, a new version of YOLO, YOLOv8, has been released. This version was even more accurate than its predecessors. On top of that, YOLOv8 had a built-in object detection model conversion from its YOLO PyTorch model to ONNX or TensorFlow Lite models, which is very useful in implementing the models on mobile applications. Therefore, YOLOv8 was chosen as the model to use for the new product.

# Setting up YOLOv8

A virtual environment for YOLOv8 is set up using the published open-source YOLOv8 algorithm on GitHub and the Anaconda distribution of Python as seen below. Various other required or useful programs were also installed to prepare for the later part of this project.



Simple Image Download is a tool that makes getting images from Google easier. Using this, a large amount of image samples were collected.



Another application used in this project was lavelImg, a tool that allows annotating images for YOLO to train from. The application will be showcased later in this document.

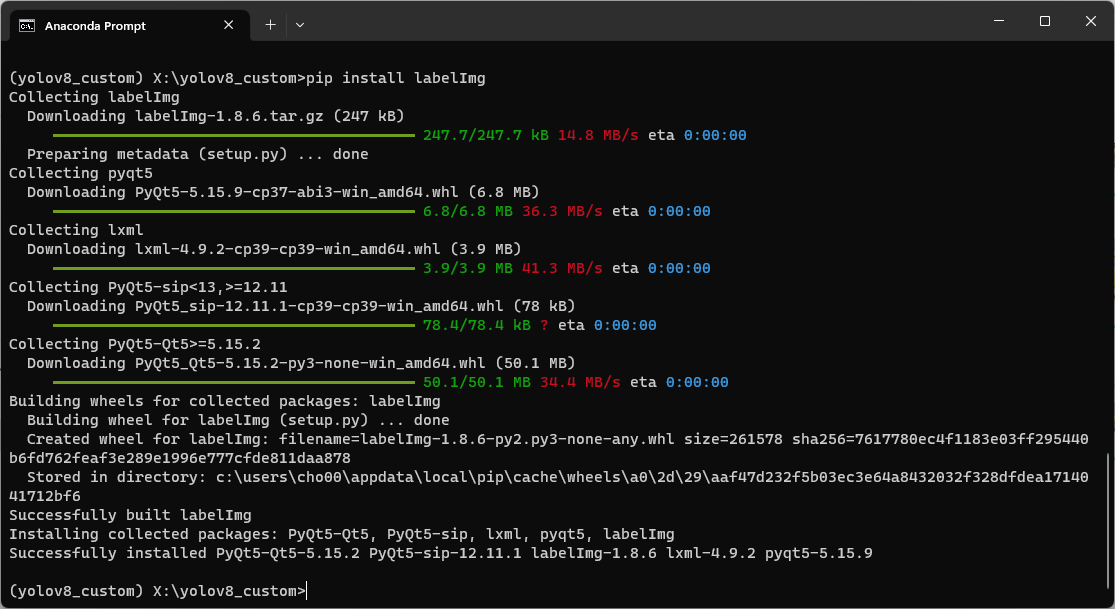
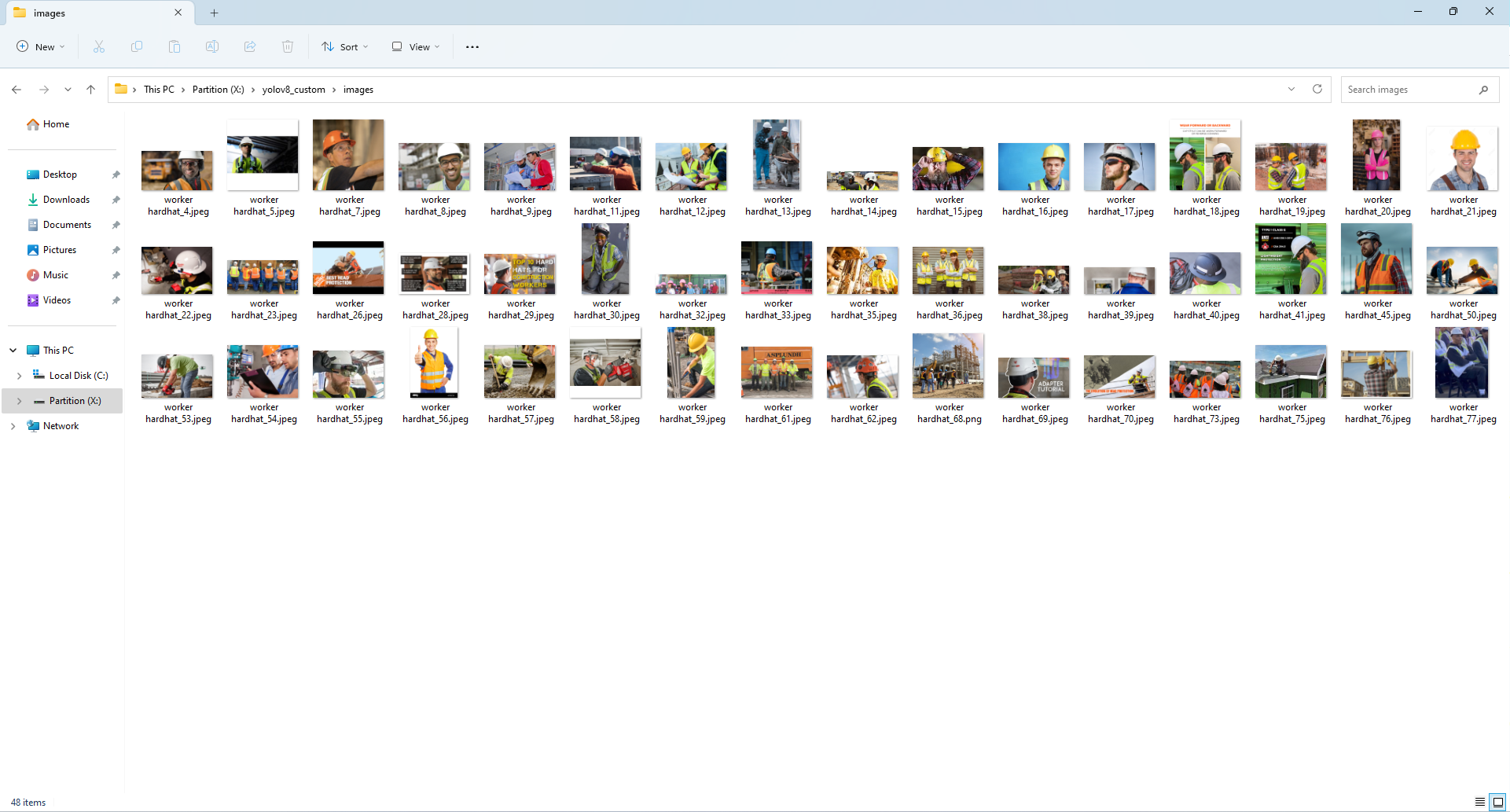
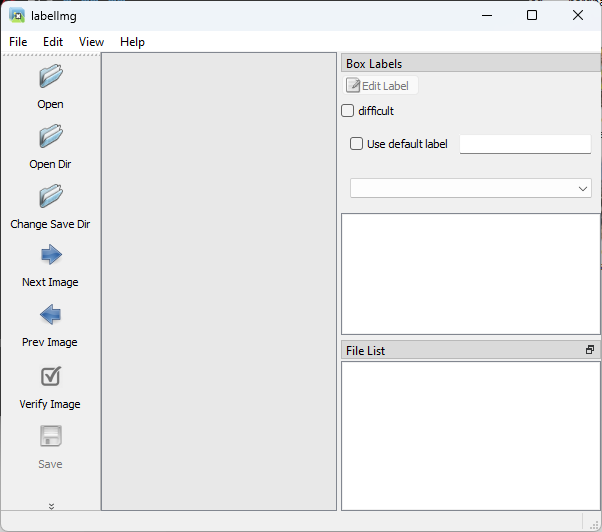


Image Annotation

Various objects are already available to be detected by the YOLOv8 model. For the objects that it cannot detect, a custom model could be made, allowing it to detect them. In this project, the objective is to detect hardhats. First, to allow this algorithm to detect custom objects, a large sample of images containing the object has to be gathered. This was done using the tool mentioned prior. Since a lot of the images were cliparts or random images that were not relevant, the sample was sorted through and got rid of the useless images.



The next step is annotating the images gathered. Another program called labelImg, downloaded from the Anaconda prompt in the previous stage, will be used to annotate the images.

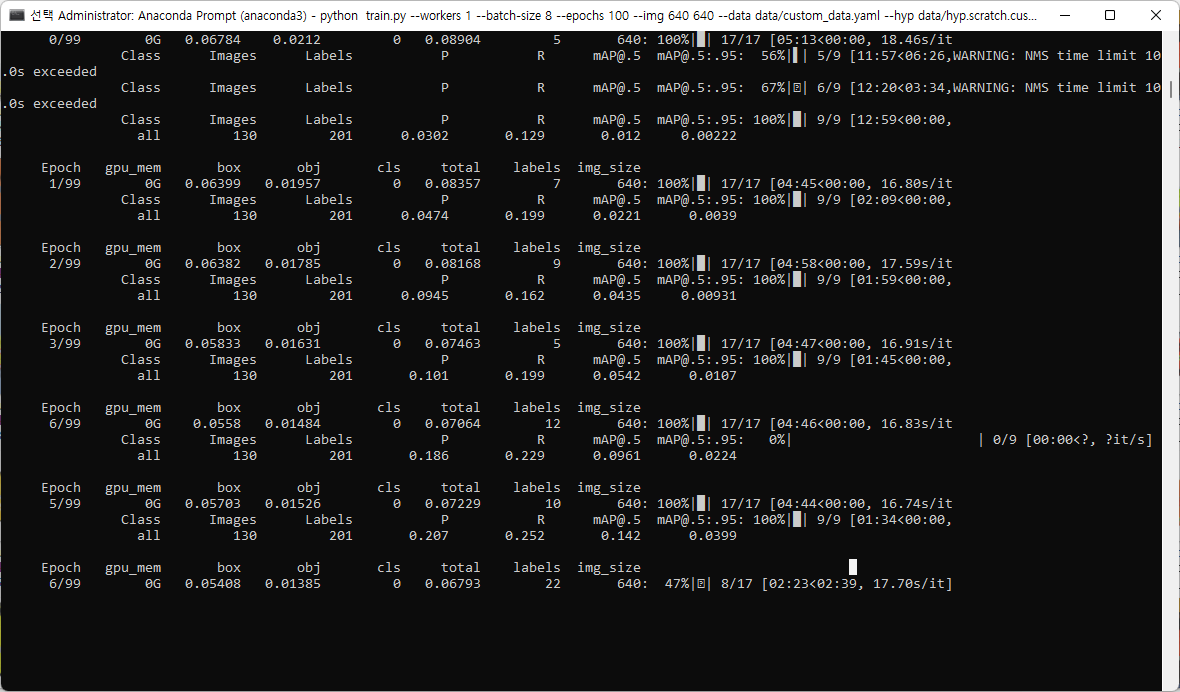


Below is what the annotating process looks like.

 Around 100 images were annotated for this project. For the model to be more accurate, more images should be annotated to be used in training. However, that would consume significantly more time, and per Mr. Mukherjee’s opinion, 100 is more than enough for such projects.

# Custom Object Detection

Using the annotated set of images, a custom object can be created. This will allow artificial intelligence to detect that object. Using the annotated set of images created from the prior step, the YOLOv8 model was able to train itself and create a custom hardhat detection model. From the experience of trouble caused by attempting to run training on a computer without CUDA core GPUs when training a custom YOLO model during Original Work, it was made sure that the training was done on a computer with CUDA cores available.



# Mobile Implementation

# Initially, the plan was to create a new Android mobile application that implements the hardhat detection model created above. However, building a refined application is not only difficult but also inefficient. The creators of YOLOv8 also created a mobile application called Ultralytics, which could run the YOLO object detection model when uploaded with datasets. This was used instead, as it was a more efficient and practical implementation while saving resources to make the model itself more efficient.

# Completion Summary

With this implementation, it is possible to use object detection even with low hardware requirements. Hardhat was chosen to be trained as it was the most crucial PPE in the construction environment. As mentioned in the first part, in the future, such a model could be used inside surveillance cameras of construction sites and warn workers when PPE equipment is not proper, increasing their safety.