Design Verification Presentation Video

You will verify to the teaching team that your design satisfies the problem you identified in the first and second milestones.

Topics of discussion should include a description of your design and how it satisfies the identified problem and satisfies your design specifications.

* Design verification may be achieved though calculations, computer simulations, prototypes, and so on.
* The presentation should be cohesive and to receive marks everyone has to contribute to the discussion.

# Controls Topic Discussion – Video Speaking Notes

Prototypes:

* A few prototypes were created over the course of the term, the prototype started with a dataset of about 500 images to train the model part of, with this number of images, the model was not able to accurately train a good model to ensure the accuracy rates that we wanted, to fix this issue, more images were added to the dataset, where we currently have about 5400 images to train throughout 8 different classes with the images sorted into a few different classes and categories.
* The current prototype has an accuracy rate of about 85-90% on most common households items, but will mis-identify the class of an item if the lighting is poor or not centered in the camera field of view. To correct this problem and to always ensure we meeting the 75% accuracy constraint more images should be added to the dataset of common household recycling products to create a more educated model.
  + If the lighting has a large reflection or glare on the items, it can think the item is something other than the item specificized.
* The current model is able to classify about 75-90 images per second from the camera, but as the model size increases the number of images the jetson can handle in a second will be reduced. A larger model of about 20,000 images was tested and the jetson was able to classify about 25-35 images per second which is still considered fast with respect to the hardware onboard the Jetson Nano.
* Another constraint of the system was that the system must be able to identify and sort at least 6 items in a minute which means that each item should be processed though the system in under 10 seconds. With the current control strategy, the jetson will continuously be processing images, if an image is detected other than the background image, the jetson will verify that the object is one of the classes by checking 10 continuous images to ensure that the object is the same class and that the certainty rating for all of the images is above the threshold of 75% as stated in the constraints. After ensuring that an object is there and selecting what bin to sort it to, the Jetson will send the signal to the motors and actuators to sort the item to the proper bin.
  + See Flowchart for control strategy on screen.
* Another constraint was that the system must not misidentify glass/Styrofoam/trash into the wrong bin more than 10% of the time. To resolve this issue, glass and Styrofoam were each created to be a class to allow the best chance of the Jetson to determine what type of item it is. To ensure that we do not sort garbage or Styrofoam into the recycling bin, the jetson will sort any item without an 80% certainty value into the garbage, because it is most likely not recycling as the Jetson did not recognize it, as well as it is better to place a recycling item into the garbage then to place a garbage item into the recycling. If the garbage item is placed into the recycling bin, then this can lead to contamination of the recycling which sometimes results in the entire recycling pile being trashed at the recycling centre.
* The final objective is that the system should be open source. All of the code and code modules that we have used for the testing and development stages for the jetson were provided from GitHub. The code that is planned to be developed to run the open-source training and models code will be placed onto GitHub as well to support the opensource nature of the project.

Please see video demo in the folder, this will be helpful and should be able to create or take parts off of it!