**Phase 1 – Data Generation**

Conclusion

* 130 images generated
* Graphical user interface

  Description automatically generatedGenerally the app functioned like it should and made capture, storage and transfer of data from a production facility in Iceland to Denmark easy and very fast. The problem was that there was not enough available fruits to take pictures of but when the fruit came the system worked very well.

Discussion

* Not enough fruits

**Phase 2 – Modeling**

* CNN results
* LCC results
* Segmentation implementation - SelfiSegmentation
* Yolo Results

Discussion

* Other possible models?
* Should try on strawberries?
* Preprocessing, why did we choose what we did and are there some things that could be improved
* Different background tests

**Phase 3 – Iterations**

* Prototypes (CNN: time + push)
* Prototype (CNN/YOLO: Sensor + push)
* Prototype (Yolo: coordinates + push)

Discussion

* Performance comparison of all 3 prototypes
* Further work / future implementations
* Pros and cons of working with Build,measure,learn cycles
* Lego as a prototyping tool / general performance of equipment

**Final Discussion**

* How to further develop the system, does it actually work in a production environment?
* Can Humble use the current prototype ?
* Limitations (Fruit bad on the other side)
  + Other types of fruits in current system
* How to implement on a real world equipment? (transition from prototype(lego) to actual product
* Feedeing mechanism
* How to clean the machine
* Threshold completely bad, very bad, little bad ,Ok, perfect
* Other use cases for the algorithm
* Lego as a prototyping tool / general performance of equipment

**Final Conclusion**

* Is this a proof of concept?
  + Yes it is and we managed to come further than we believed originally
* Did the project realize the initial goals