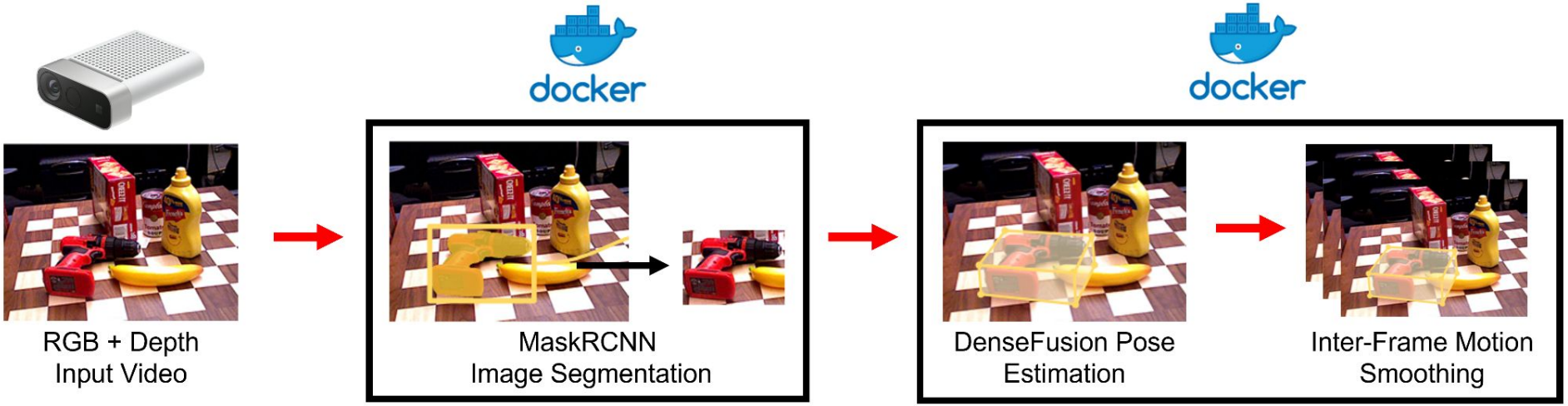
Group 5 - Pose Estimation

Nikolas Lamb, Gurpreet Kaur, Houchao Gan, Mingjun Li,
Noah Wiederhold, Priyo Prosun, Stephen Miner

Reviving Pose Estimation Using Docker Containers

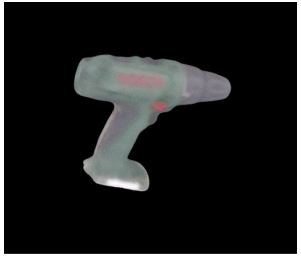


Existing pose estimation approaches are problematic to install on modern hardware. Our project standardizes the installation process and provides containerized environments to make setting up a pose estimation process quick and easy. We provide code at: `nikwl/DenseFusion`

Azure Kinect Integration

Record color + depth live

Segment color + depth to focus on object



Add pose prediction into moving average of last *n* poses/frames

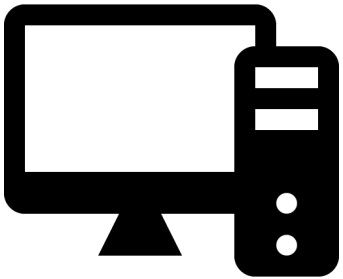
Predict pose of object

rotation R_i

translation t_i

confidence c_i

Containerized Pipeline



Kinect
Main PC

Save

Mask RCNN
Docker

Save

Densefusion
Docker

3D transform
video


Soft Scrub
Domino
SUGAR
Tomato Soup

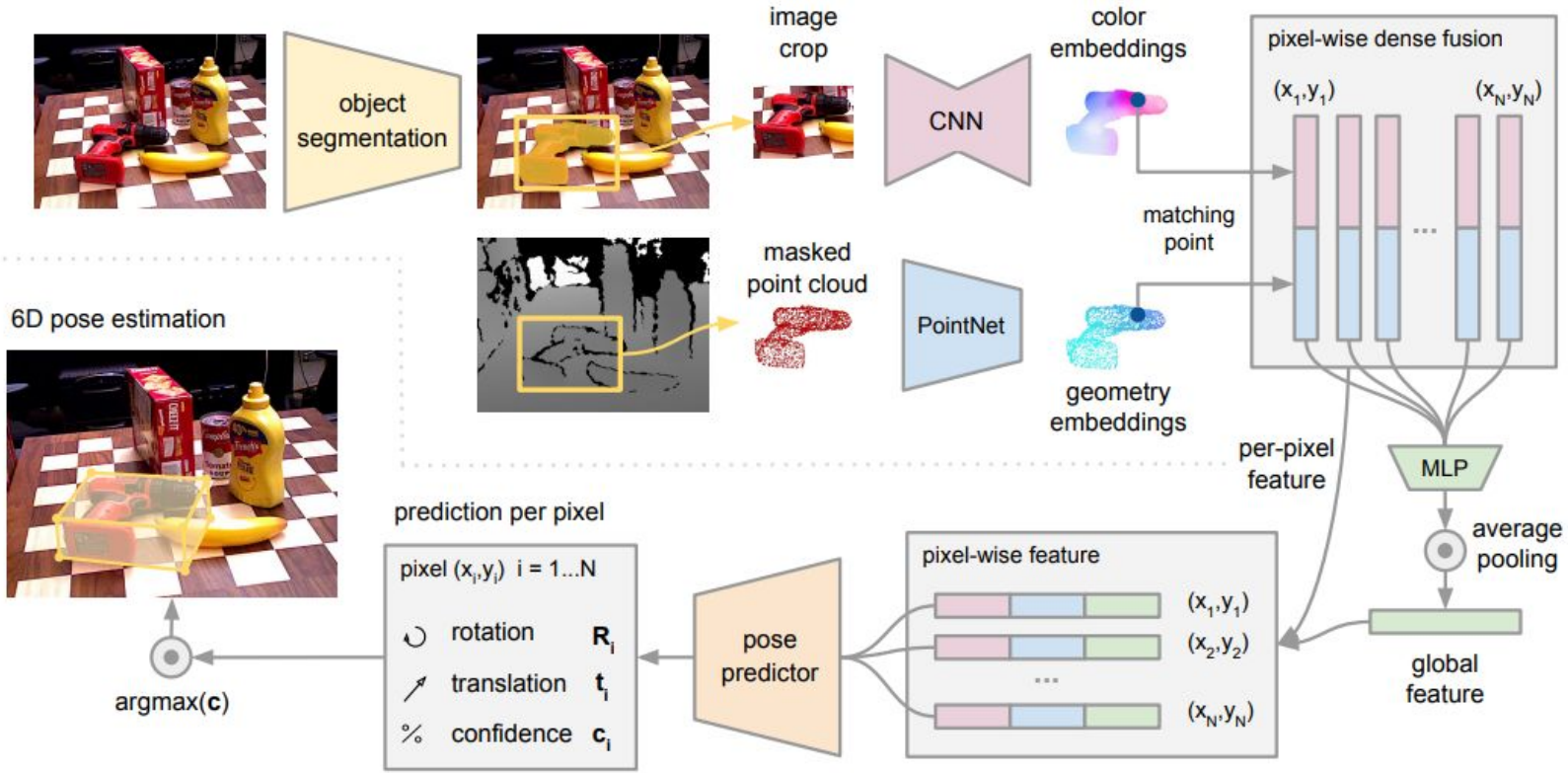

Domino
SUGAR


CHEEZ-IT

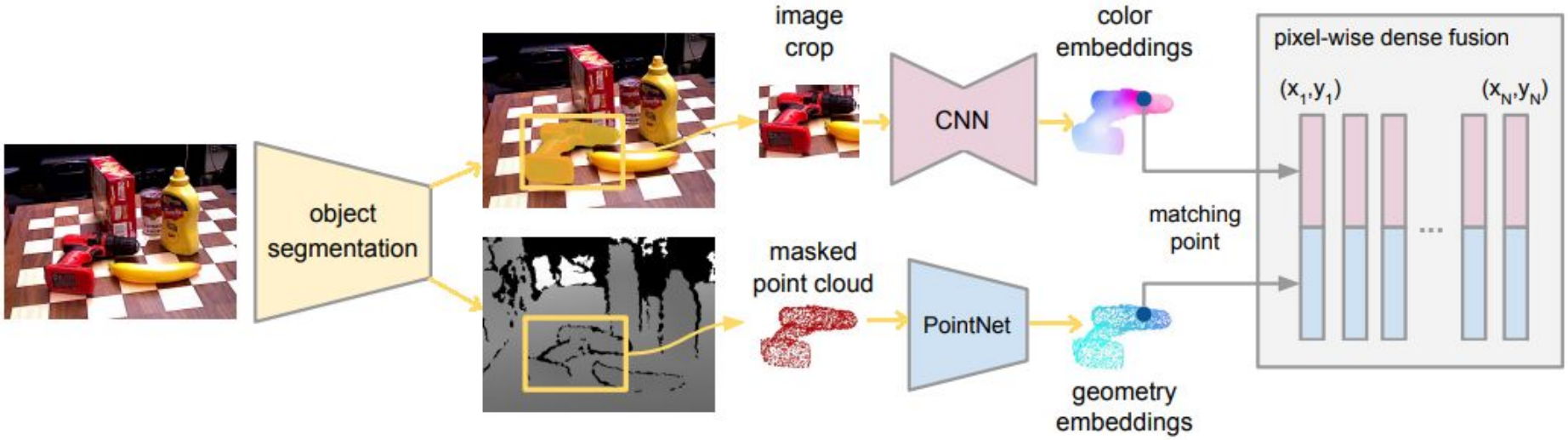

CHEEZ-IT


LITHIUM

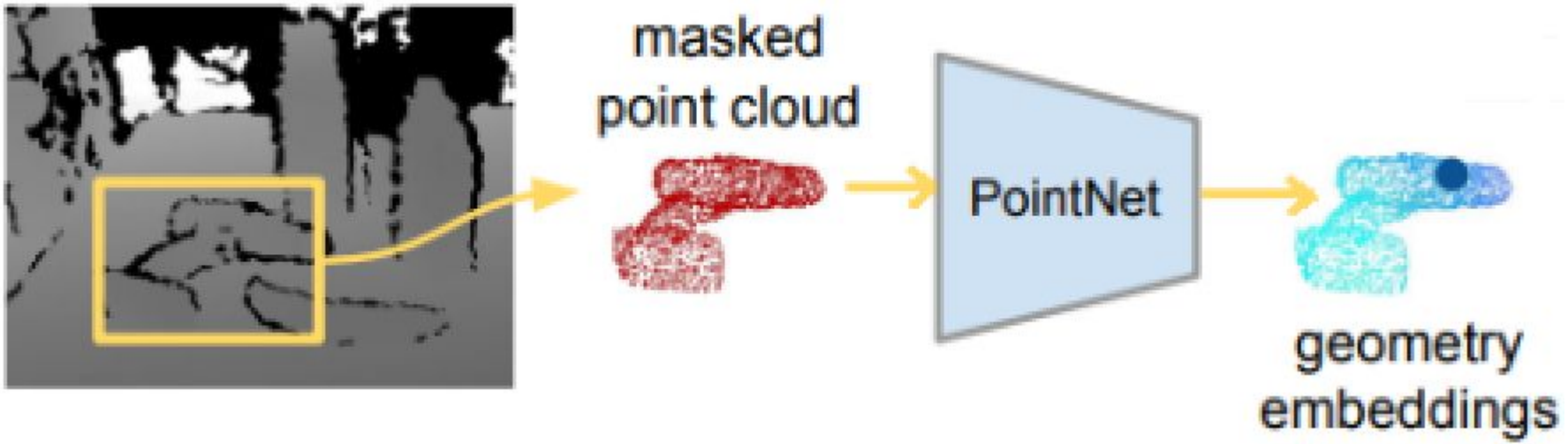
Network Modifications



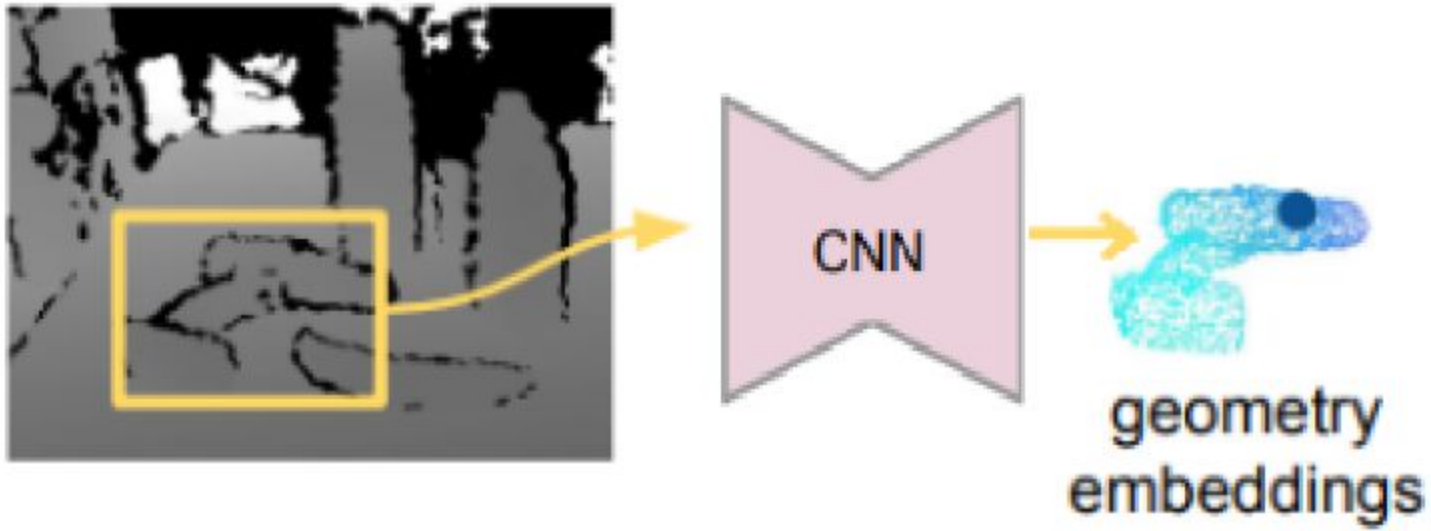
Current pre-processing



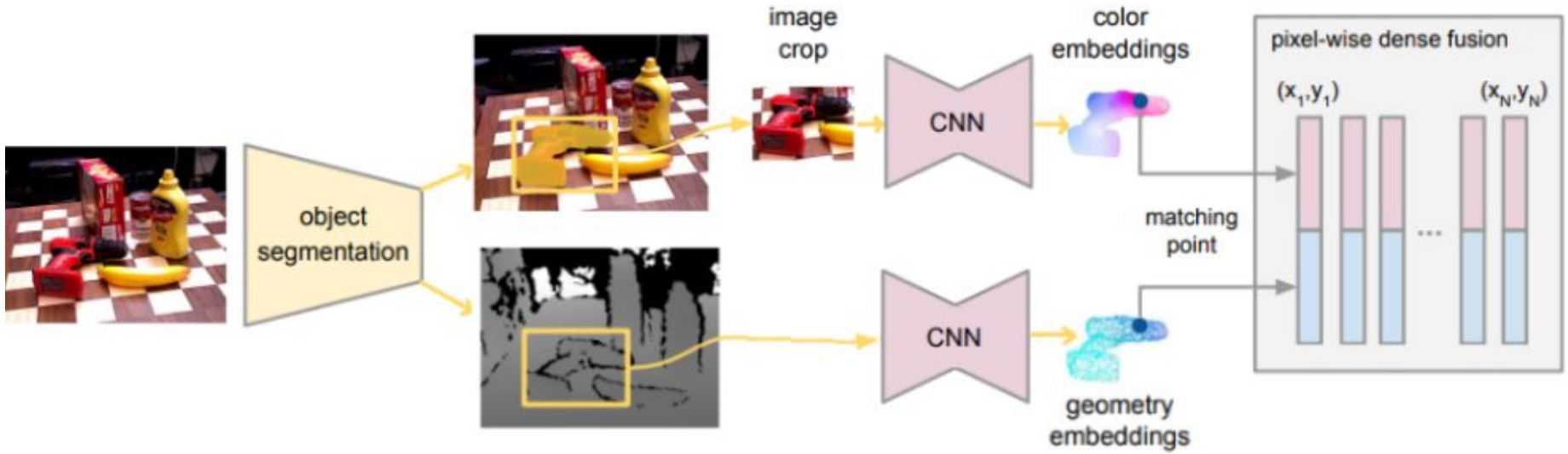
- Computationally costly
- Seems unnecessary to convert to point cloud in order to get the embeddings



- Much less complexity
- Same type of output to use as input for dense fusion

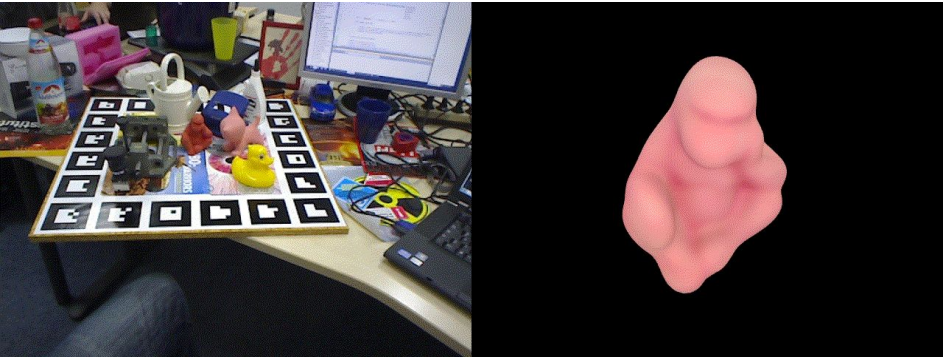


The change in context

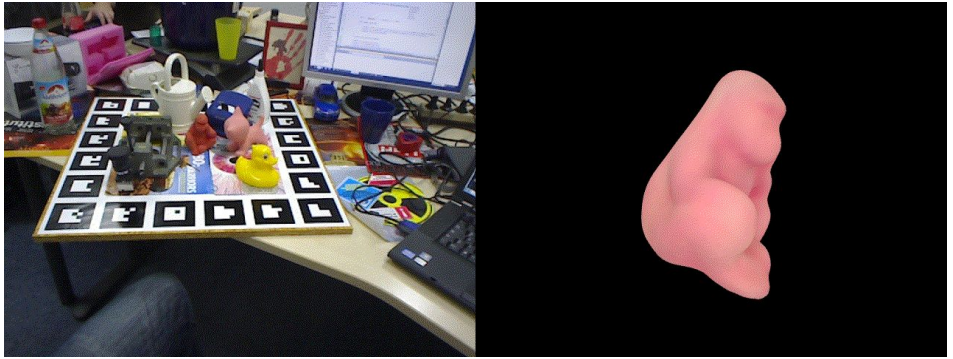


Training results

Results from original training with points cloud



Results from training with depth image(25 epoch)



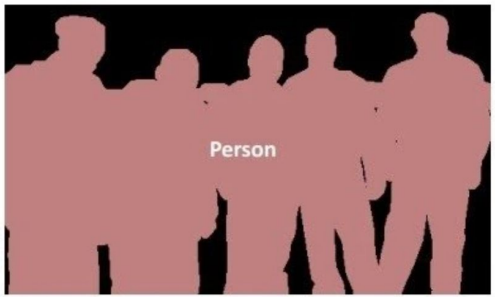
10 epoch

Tasks

- Noah, Nik, Mingjun:
  - Get DenseFusion training (fix batching)
  - Live prediction working with static mask (waiting on segmentation)
  - Motion smoothing
- Gurpreet & Priyo: Get segmentation network working and integrated into pipeline
- Houchao & Stephen: Modifications to DenseFusion to use depth map inputs

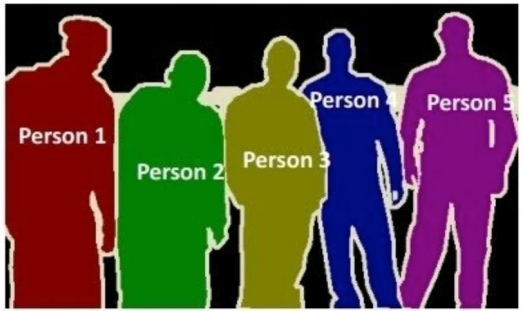
Semantic Segmentation Vs. Instance Segmentation

Classifies each pixel into a fixed set of categories without differentiating object instances.



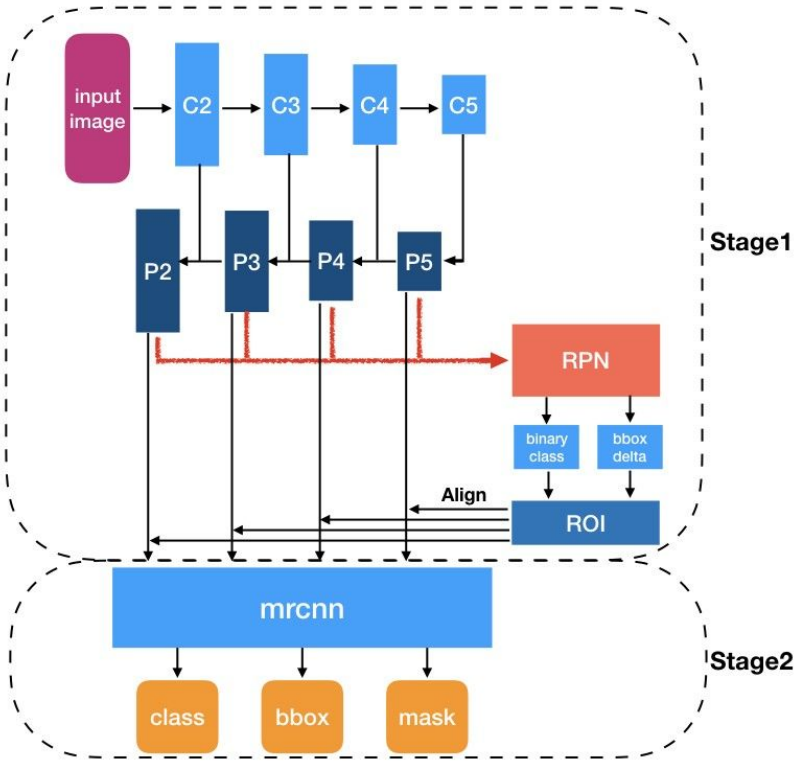
Semantic Segmentation

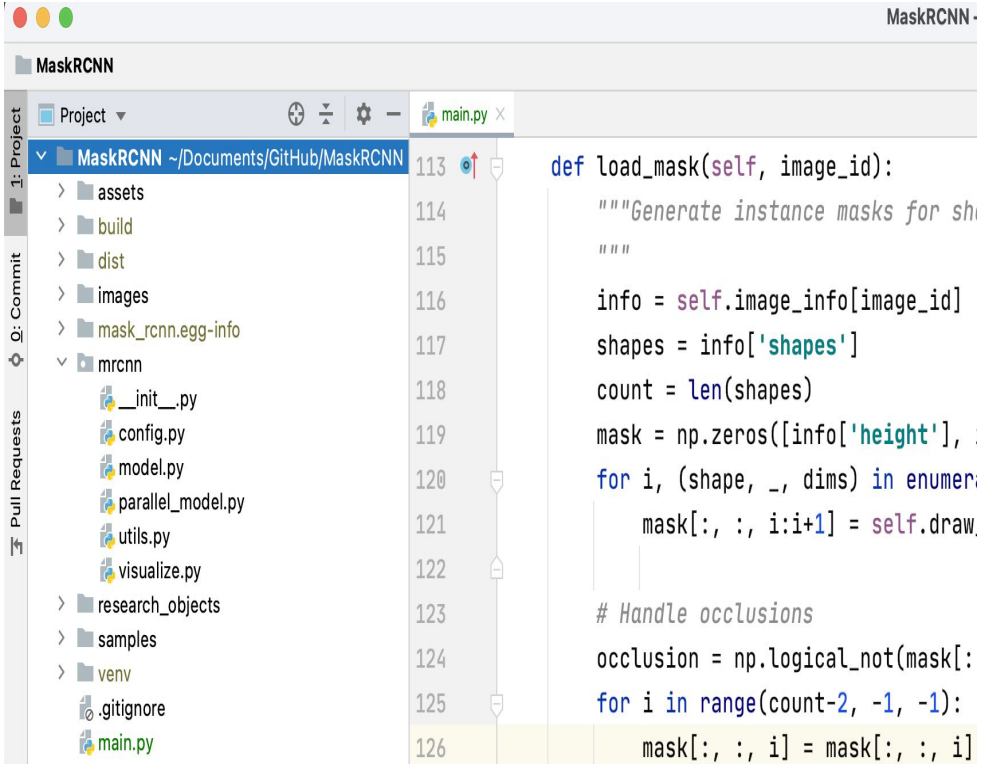
It deals with the correct detection of all objects in an image while also precisely segmenting each instance.



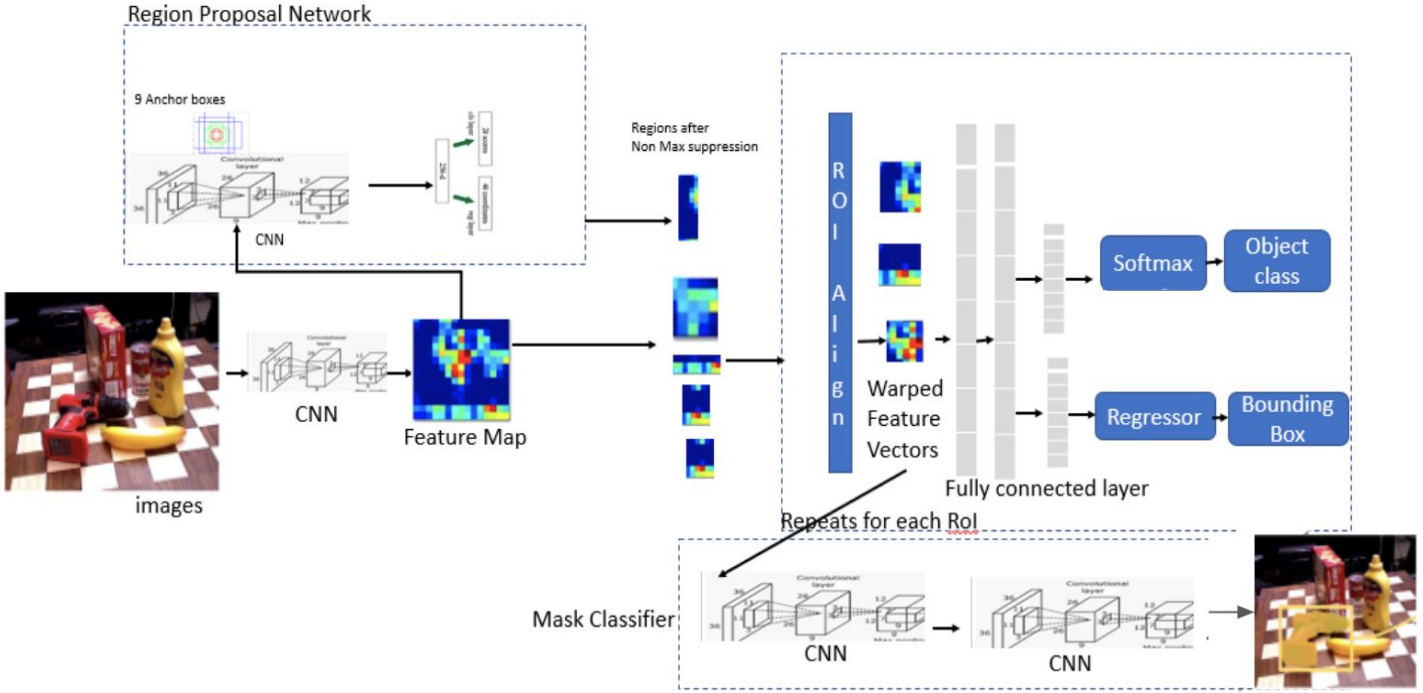
Instance Segmentation

Why Mask RCNN?





Mask RCNN


Region Proposal Network
9 Anchor boxes
CNN
Regions after
Non Max suppression
images
CNN
Feature Map
ROI Align
Warped
Feature
Vectors
Softmax
Object
class
Regressor
Bounding
Box
Fully connected layer
Repeats for each RoI
Mask Classifier
CNN
CNN

What have we produced?

Trainable densefusion Docker image:

- `docker run -it nikwl/densfusion:latest bash`

Trainable pixel2mesh Docker image:

- `docker run -it nikwl/pixel2mesh:latest bash`

Trainable UNET architecture.
Pose visualization suite.

(partial) Integration with azurekinect sdk for real time prediction.