6-D pose and dimension estimation of objects

Anju S

Indian Institute of Space Science and Technology(IIST)

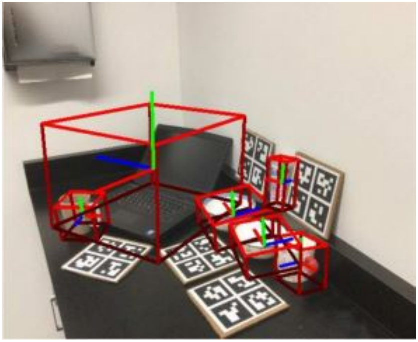
anjuskumar1313@gmail.com

August 6, 2020

Overview

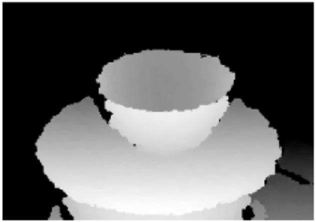
Introduction

- 6-D object pose estimation deals with the **3D position and 3D rotation** of objects in camera-centred coordinates.
- Gives promising solutions for problems in scene understanding, augmented reality, control and navigation of robotics etc.



Two streams of methods are usually used for 6D pose estimation:

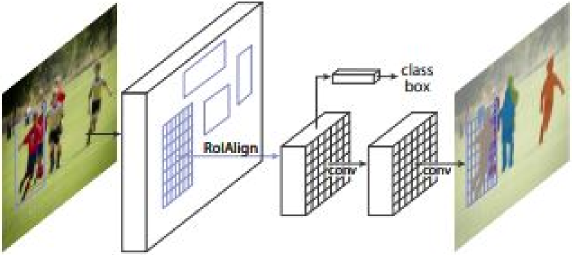
1. From RGB images: PnP(Perspective-n-point) algorithm, Fiducial Markers etc.
2. From RGB-D images: Depth data gives us the 3rd dimension.



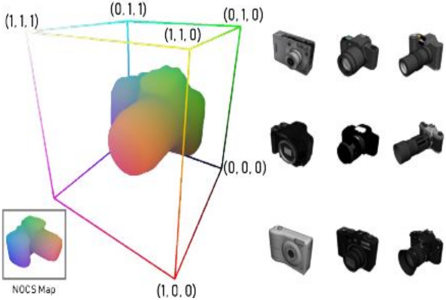
We will be focusing on 6D pose estimation from RGB-D images.

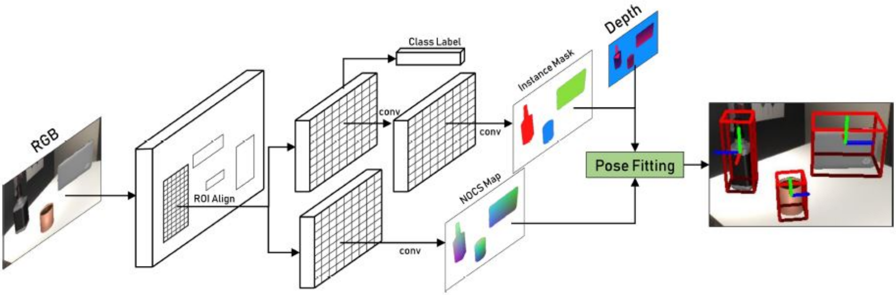
Model

- The base of our model is the **Mask-RCNN** network.
- Mask-RCNN provides us with
  instance segmentation and classification.



To the Mask-RCNN network we add one more convolution block which is trained to output an NOCS mapping for every object.





- From instance masks and depth data we obtain a 3D point cloud which is compared with the 3D point cloud obtained from NOCS maps to obtain the rotation and translation of each object.

Umeyama algorithm

Let P and Q be the two 3D point cloud data obtained.

- Subtract the respective centroids from both P and Q to bring them to the common center.

$$A = \frac{Q.P^T}{n} \tag{1}$$

$$UDV^T = SVD(A) \tag{2}$$

- Rotation = V.U^T
- Scale Factor(SF) = $\frac{Trace(D)}{Sum(X,Y,Z)}$
  Scale = [SF, SF, SF]
- Translation = Centroid(P) - Centroid(Q).(SF x Rotation)

Implementation

The network has been trained with two different datasets:

1. CAMERA dataset(Context Aware Mixed Reality Approach):
    - 6 categories of objects(bottle, bowl, camera, can, laptop, and mug) are synthetically placed in real background with different lighting conditions.
    - It consists of 275K training and 25K testing images.
2. REAL-world dataset:
    - It consists of 4300 training, 950 validation and 2750 testing images.

The training is done in 3 stages.

Stage-1: The network head layers are trained for 100 epochs at a learning rate of 0.001.

Stage-2: Layers above stage 4+ are trained for 130 epochs at a learning rate of 0.0001.

Stage-3: All layers together are trained for 400 epochs at a learning rate of 0.00001.

Loss function

- The loss function used for bounding box regression is a commonly used one called **Smooth L1 loss**.
- Smooth L1-loss combines the advantages of L1-loss (steady gradients for large values of —x—) and L2-loss (less oscillations during updates when —x— is small).

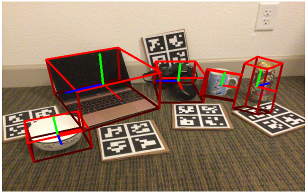
$$L1LossFunction = \sum_{i=1}^{n} |y_{true} - y_{predicted}|$$

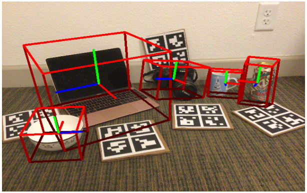
$$L2LossFunction = \sum_{i=1}^{n} (y_{true} - y_{predicted})^2$$

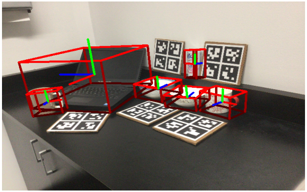
$$L_{1;smooth} = \begin{cases} |x| & \text{if } |x| > \alpha; \\ \frac{1}{|\alpha|}x^2 & \text{if } |x| \le \alpha \end{cases}$$

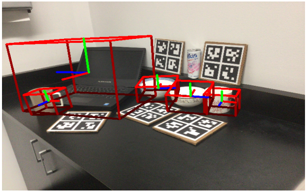
Testing results

These are the final results obtained after testing the model on the real world testing dataset.





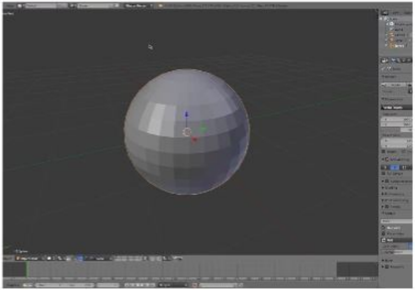




Hand Dataset

- Our current problem concerning the **humanoid** project requires the identification of the 6D pose of the **hand of the robot** in various orientations and lighting conditions.
- The current network does not have hand as one of its classes ie. it has not been trained to identify a hand. Hence we need a **dataset of hand** images and corresponding ground-truths on which the network needs to be **train**ed.

- We have used a software called **Blender** for creating the dataset.
- It is a free and open-source 3D computer graphics software used for creating animated films, visual effects, 3D printed models, computer games etc.



Contents of our dataset :

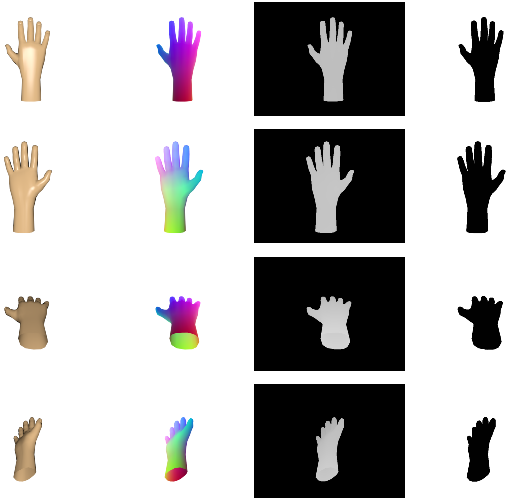
1. RGB hand images
2. NOCS maps
3. Instance masks
4. Depth maps
5. Labels

- The training dataset consists of 432 images and validation dataset has 114 images comprising a total of 91 different orientations.
- It consists of images with 3 different lighting setups.

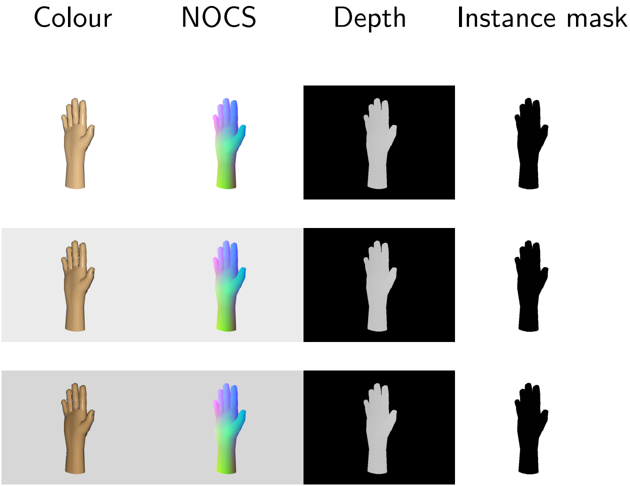
Sample images

Different orientations

Colour NOCS Depth Instance mask

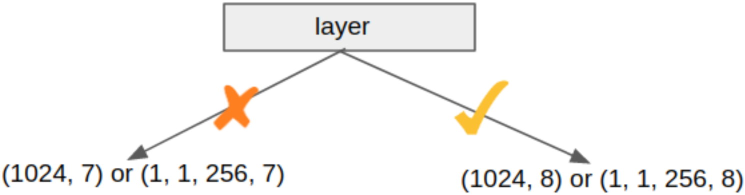


Different lighting conditions



Training

- The next step would be to train the network using these images.
- **Idea:** We will be using the pre-trained weights(6objects) as a base and then train the network using our hand dataset over it.

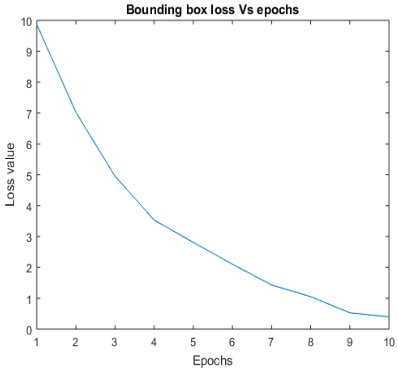


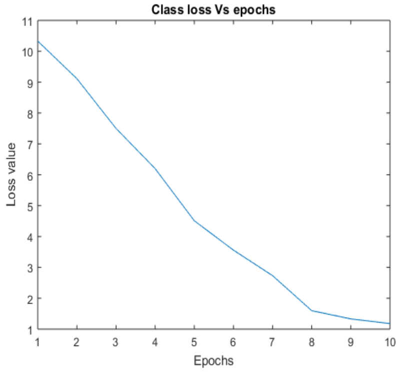
Plan:

$$(\text{None, None, 0:7})_{new} = (None, None, 0:7)_{old}$$
$$(\text{None, None, 8})_{new} = Mean[(None, None, 0:7)_{old}]$$

The network has a total of 462 layers and the above mentioned correction was made manually for every layer with shape anomaly.

- Number of epochs $= 10$
- Learning rate $= 0.001/100$





Ideas for the rest of the project

1. Hand Dataset
   - Expansion of the hand dataset by adding other possible hand poses of the robot.
   - Improving the hand dataset by addition of realistic background to the current images.
2. Addition of IoU(Intersection over Union) loss to the loss function for bounding boxes. This could improve the bounding boxes we have obtained with Smooth L1 loss.
3. Train the network using rotated versions of images so that the network becomes familiar with a wide range of view points.

References

- Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick, *Mask r-cnn*, Proceedings of the IEEE international conference on computer vision, 2017, pp. 2961–2969.
- Bugra Tekin, Sudipta N Sinha, and Pascal Fua, *Real-time seamless single shot 6d object pose prediction*, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 292–301.
- He Wang, Srinath Sridhar, Jingwei Huang, Julien Valentin, Shuran Song, and Leonidas J Guibas, *Normalized object coordinate space for category-level 6d object pose and size estimation*, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 2642–2651.
- Yu Xiang, Tanner Schmidt, Venkatraman Narayanan, and Dieter Fox, *Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes*, arXiv preprint arXiv:1711.00199 (2017).

Thank You