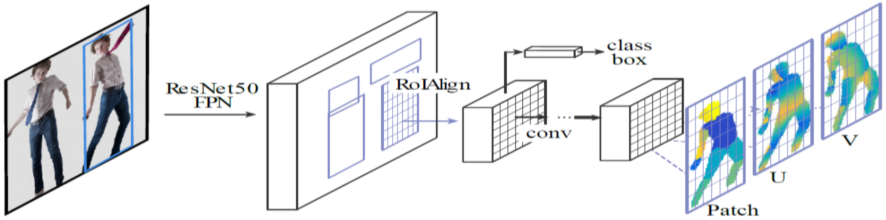
Human Pose Estimation (DensePose)

- Pravallika Reddy

Problem Statement

To perform **pose estimation** by generating dense correspondences between an image and a surface-based representation of the human body using Mask - RCNN as proposed by the authors of “DensePose: Dense Human Pose Estimation In The Wild”.

- Pose estimation deals with detection of human body parts and estimating their orientations in an image.
- The project deals with estimating the pose solely from an RGB image, hence overcoming the limitations of conventional pose estimation methods of using a depth sensor.



Approach

To correctly perform Dense pose estimation, the following series of approach is followed:

- Preparation of a ground truth “image to 3D surface mapping” **dataset** to be fed as the training data to the supervised Machine Learning model.
- Reducing the search space for point correspondence by generating **region proposals**.
- **Object Detection and Semantic Segmentation** - performing classification of the ROIs into 25 different body parts using Mask RCNN
- **Pose Estimation** - Getting point correspondences or estimating 3D dense position by performing Dense Regression on the ROIs.

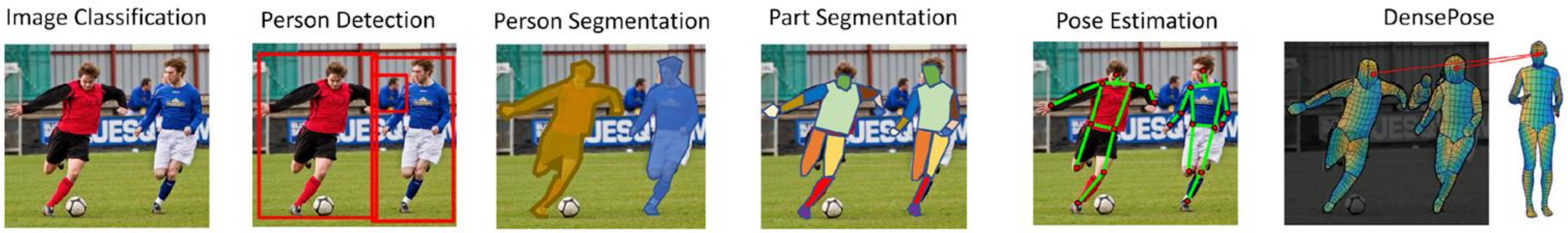
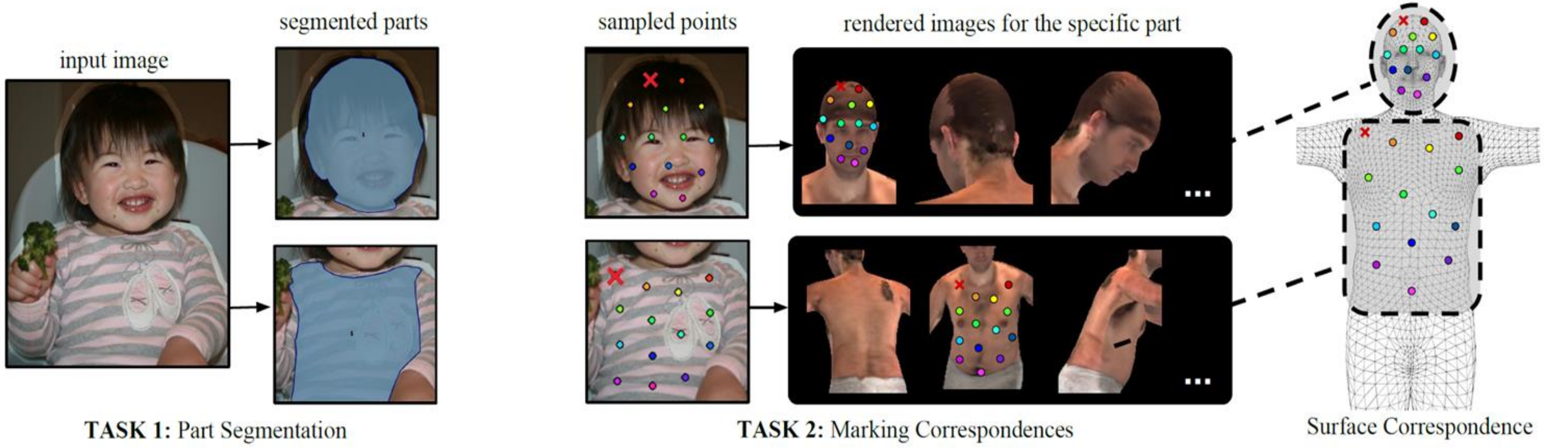


Image Source: http://presentations.cocodataset.org/ECCV18/COCO18-DensePose-Overview.pdf

Dataset



Tools Used

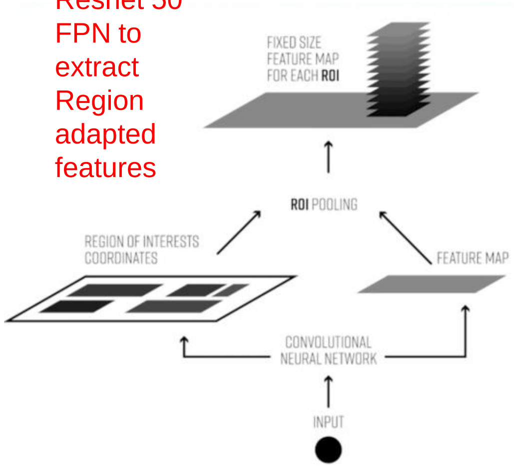
- Google Colaboratory
- Python dependencies:
  Caffe2 , PyTorch

- The dataset consists of 50K images and their manually annotated image-to-surface correspondences
- DensePose - COCO dataset
- Test image/video from personal device

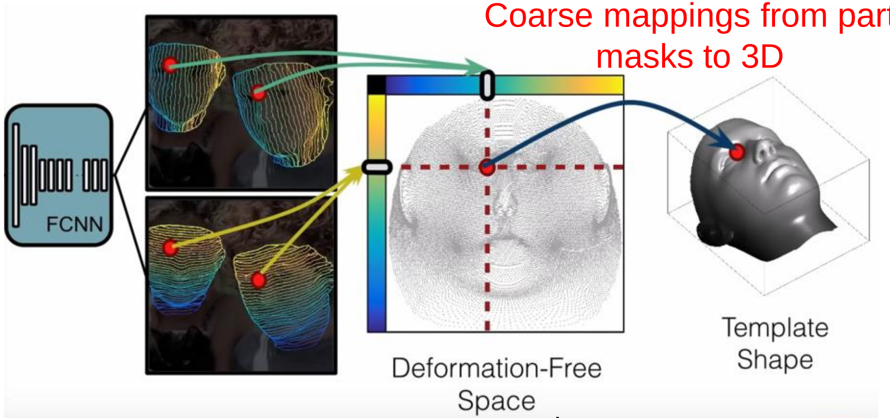
Overview

Part classification and segmentation using Mask-RCNN

ROI pooling:
Resnet 50 FPN to extract Region adapted features



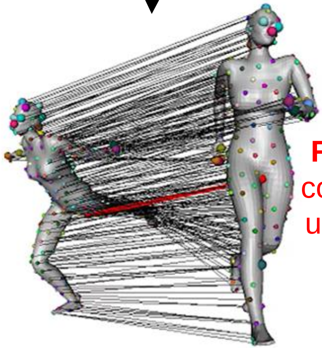
Pixels in each ROI classified in 25 segments



Coarse mappings from part masks to 3D

RGB Input Image

A mapping from image pixels into a dense template grid via a Fully Convolutional Neural network.



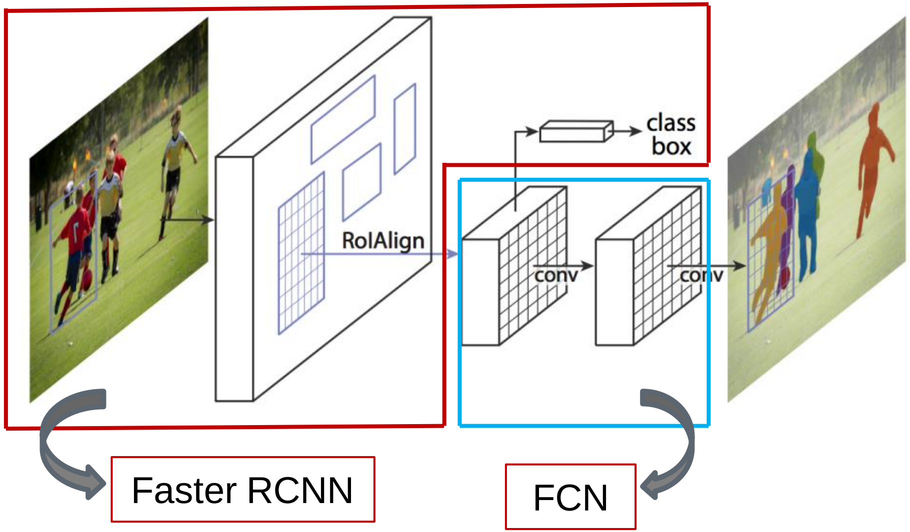
Regressing 3D correspondences using DenseReg architecture

Dense-Pose RCNN Model Implementation

Step 1:

***** Using ResNet model's pretrained weights, a mask RCNN model was trained in the **Detectron Framework** to detect and segment humans in the image.

***** Detectron framework includes Mask-RCNN, faster RCNN, RPN, object detection algorithms using Feature Pyramid Networks and ResNet as backbone network architectures.



Step 2:

***** Using **DensePose** ResNet 101 FPN architecture, the 3D point correspondences of an RGB image was inferred and visualized.

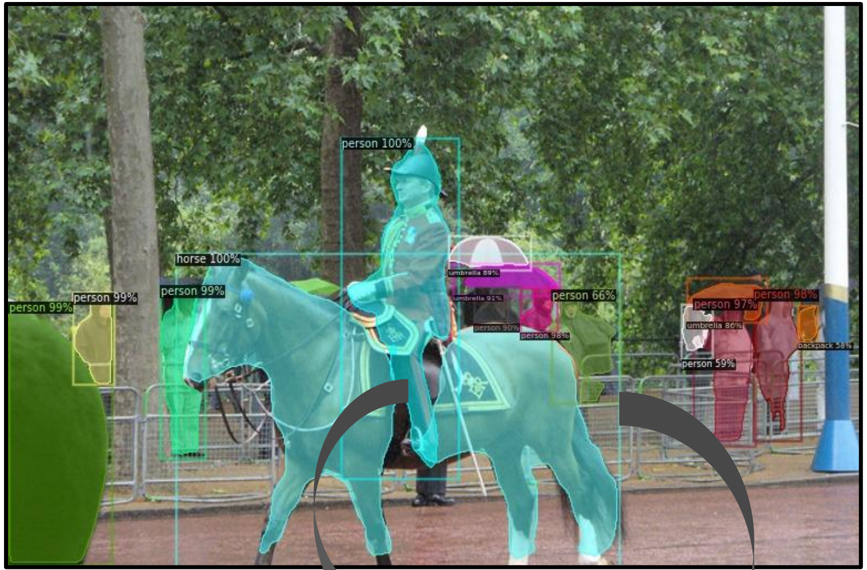
***** The DensePose architecture is built on the Mask-RCNN Detectron framework to classify and mask 24 body parts and 1 background.

***** The 24 body parts are sent through 24 regression branches that use DenseReg architecture to achieve image to 3D surface correspondences.

Results: Mask RCNN implementation

Input image (from the COCO Dataset)

Visualization after Boundary identification



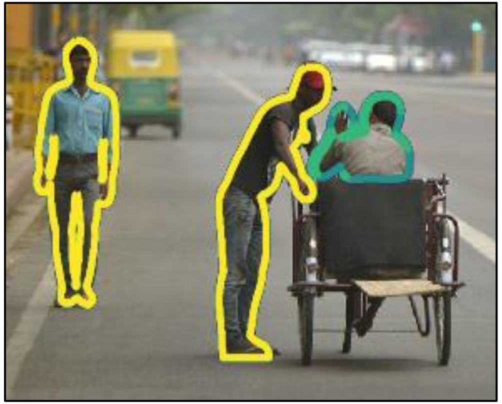
Mask

Boundary

Results: Pose Detection on Test image

Input image

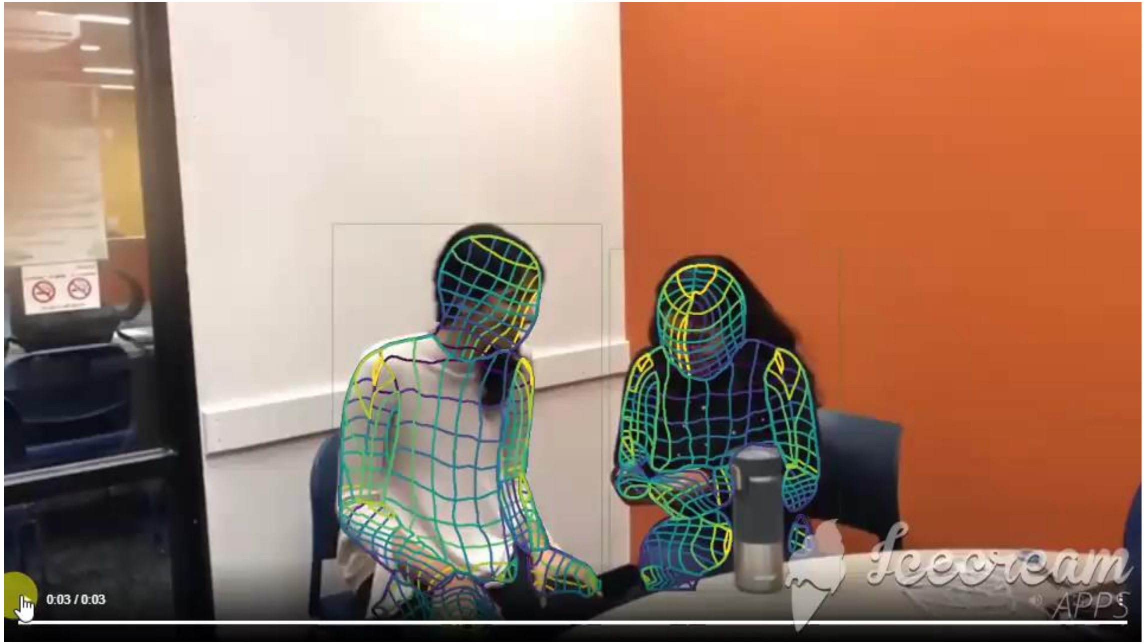
Visualized contours



Visualized mask indices

Texture transfer

Results: Pose Detection on Video



(Play Video)

Future scope

Building DensePose onto the Smart Body Measurement project to make a Virtual Dressing Room Environment.

3D animation becomes more convenient by replacing present Motion capture techniques.

Thank you!