Pose2Seg

Detection Free Human Instance Segmentation

Team : Feature Detectors

Tushar (2018102041)
Abhiram (2018102036)
Karan (2018102034)
Sachin (2018101108)

Mentor TA :

Veeravalli Sai Soorya Rao

Repo URL :

https://github.com/Computer-Vision

Section 1 : Introduction…

Introduction...

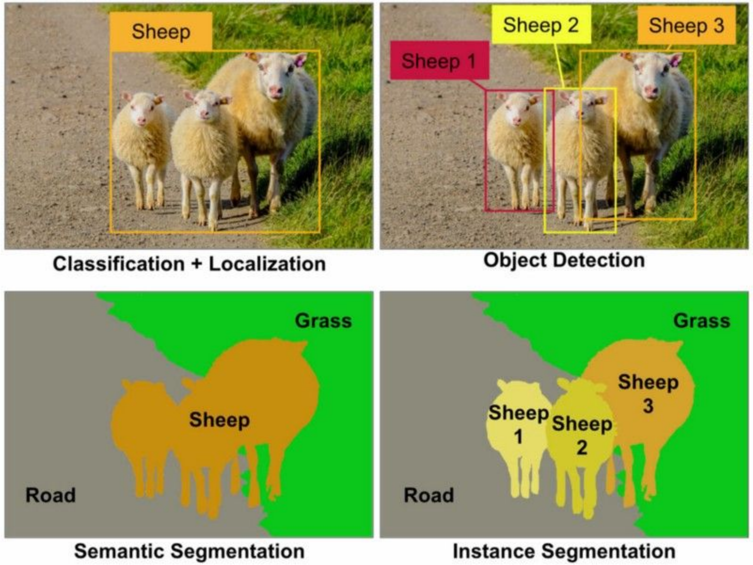
This paper addresses the problem of segmentation of Humans (sub category of object instance segmentation). The paper proposes to solve the problem using a pose-based instance segmentation framework.

The contributions of the paper are as follows:-

- A pose-based Human instance segmentation framework.
- A pose-based align module : Affine-Align.
- A segmentation module guided with artificial pose Skeleton features .
- Dataset - OCHuman with annotations (which focuses on the heavy occlusion).

Instance Segmentation

Identifying each object at a pixel level. Labels are both class and pixel aware.
It is considered the hardest problem among common use cases in CV.



Classification + Localization

Object Detection

Semantic Segmentation

Instance Segmentation

Classification+localization :

This is an image of sheeps

Object Detection :

There are 3 sheeps at these locations.

Semantic Segmentation :

There are sheep, road and grass pixels.

Instance Segmentation :

There are 3 different sheep at these locations

In this paper we discuss the special case of Human instance segmentation

Existing Solutions...

General method for instance segmentation:

1. Object detection
2. Segmentation from bounding box

Mask-RCNN based methods perform these two steps jointly.

Instance segmentation using pose information:

Top down methods:

1. Object detection (bounding box)
2. Single person pose estimation on each human instance

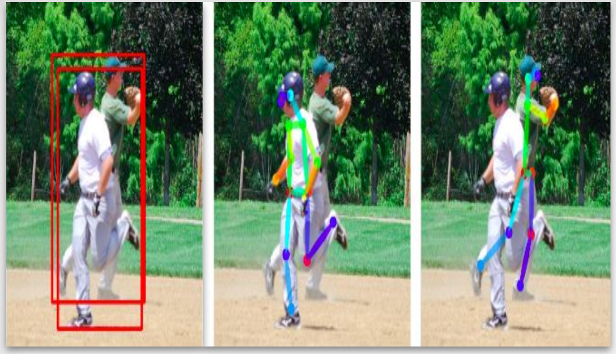
Bottom up methods:

1. Detect key points for each body part for all the people.
2. Clustering key points to form different instances of human pose.

Problems with existing Solutions...

Pipeline for Fast/Faster RCNN, YOLO, etc.

1. Generating proposal regions
2. Non-maximum Suppression (NMS)
3. Segmentation :
   a. Bounding box
   b. Pixel Segmentation
4. Classification : SVM



Problems with NMS (stage 2):

NMS can potentially reject useful bounding boxes in Heavy occlusion cases.
Pose skeletons prove to be better descriptors of Human instances.

Section 2 : Pose2Seg

Procedure

The paper divides Detection Free Human Instance Segmentation into :

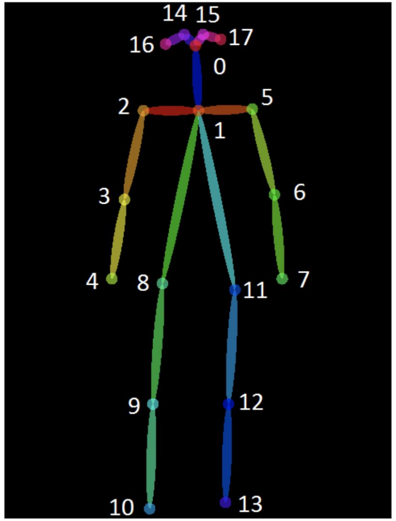
- Affine-Align Operation
    - Pose Representation
    - Pose Templates
    - Estimate Affine Transformation Matrix
- Skeleton Features
- SegModule

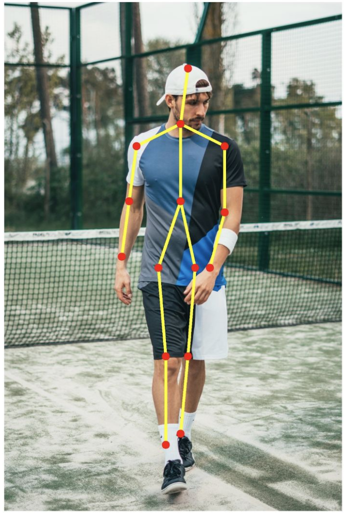
Affine align
Pose Representation :

Human Pose is described in COCO and OCHuman as :

- 17 points - one for each joint
- 3 coordinates per point - (x,y,v)
- R^{17x3} vector

Notation : x,y are normalised coordinates - (0,1)
if C j is visible (x, y, 2)
if C j is not visible C j = (x, y, 1)
(0.5, 0.5, 0) if C j is not in image





Human pose representation :

There are 17 joints (as in COCO- our training dataset), each joint has 3 coordinates - (x,y,v) which are for position in image and visibility. So, each pose is represented by R^{17x3} vector.

Distance metric used for K-means is euclidean distance.
We only consider poses with more than 8 valid points for K-means clustering.

Mean vector of the class after clustering is taken as the representative element of the class (class template) which is used for pose detection in future.

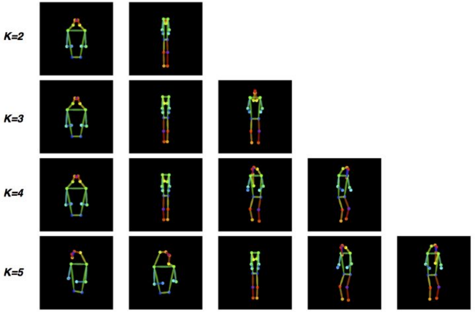


Figure 5: Pose templates clustered using K-means on COCO.

Affine align

Pose Templates :

Training the K-means clustering algo for pose templates:

1. Crop and resize the ROI into unit square, estimate the pose vector (R^{17x3})
2. Apply K-means clustering on R^{17x3} vectors.

Distance metric for K-means :

Euclidean distance.

Threshold for data points :

Poses with more than 8 valid points are considered.

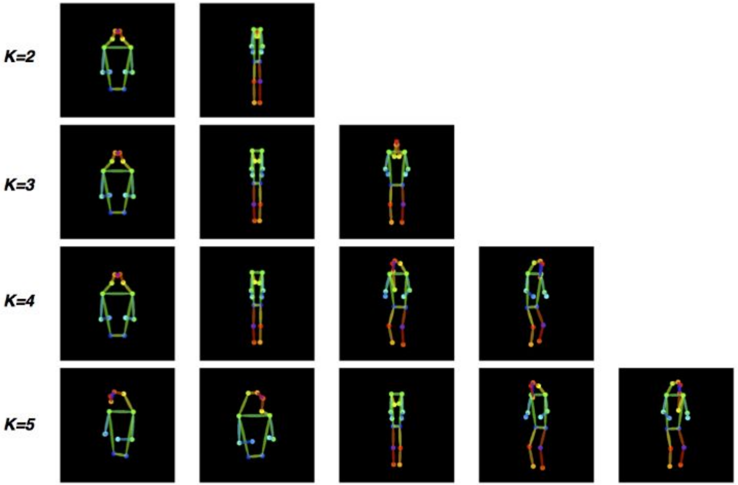


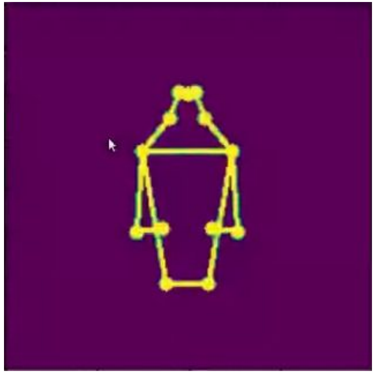
Figure 5: Pose templates clustered using K-means on COCO.

Experimenting on different K-values

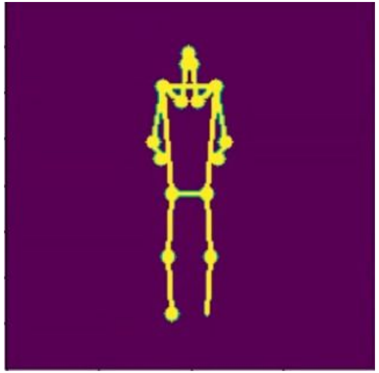
Finally k = 3 is chose for our purpose

Class means are taken as the post templates

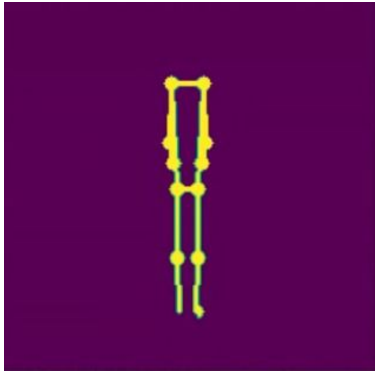
When K ≥ 4, the difference between left and right are introduced. Since our align process copes with the left-right flip, K ≥ 4 seems un-necessary for our framework. So finally, we choose K = 3 to cluster pose templates in our approach.



Half - Body Pose



Full Body Front



Full Body Back

Affine align
Best Pose Template:

1. Find an affine transform that best fits each of the template poses.
2. Select the template pose with the best score.

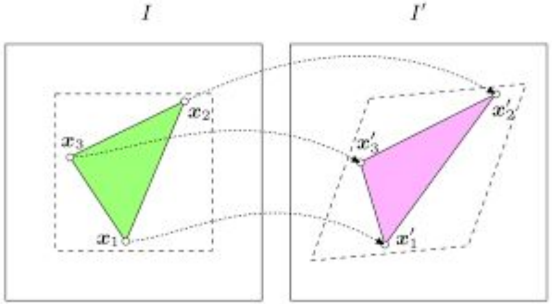
Finding affine transform:

$H_{2 \times 3}$ is an affine transform for a 2D image. Find the matrix H as:-

$$H^*_{2 \times 3} = \text{argmin}_H(|| H.P - P_\mu ||)$$

Finding score for best fit:

$$\text{Score} = \exp(- || H^*.P - P_\mu ||)$$



Affine transform

Skeleton Features:

There are 55 skeleton features per pose that we make use of while segmentation. These features are of two types

- Part Affinity fields
- Confidence maps

Part affinity fields are 2-channel vector field map for each skeleton (line that joins two joints).

There are 19 skeletons defined in COCO dataset.

Hence for each pose there are 38 channels of PAF features.

Part confidence maps emphasize the importance of those regions around the body part key points (parts = joints).

There are 17 joints defined in the COCO dataset

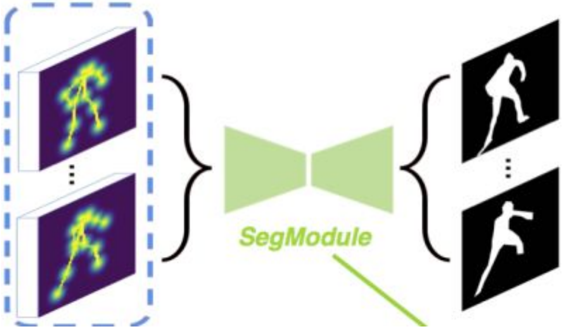
Hence for each 17 channels of part confidence maps.

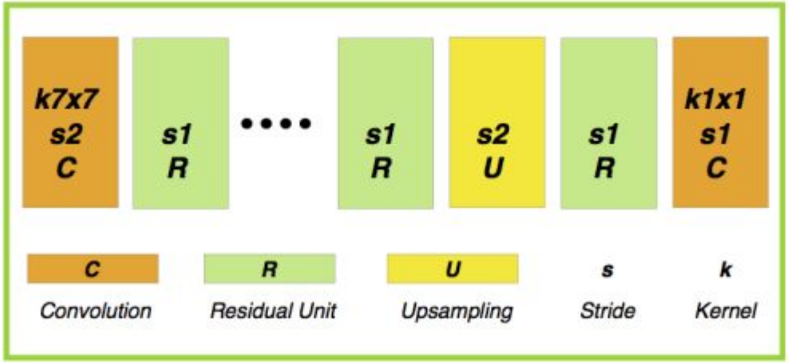
So for each instance to be segmented 55 channel skeleton features are extracted.

SegModule :

SegModule is introduced to extend the image features after alignment and is based on the resolution of the aligned RoIs.

The overall architecture is demonstrated below :





Section 3 : Dataset

OCHuman dataset

- It contains images with average IOU 0.67, it is divided into 2 parts : moderate (0.5-0.75 IOU) and hard (>0.75). It is very challenging for this reason.

- It has annotations (bounding-boxes, human poses - 17 body joints contains left and right instances of eye, nose, ear, shoulder, elbow, wrist, hip, knee and ankle and instance masks).

- It is used only for testing and validation, not for training. For training COCO dataset is used. (COCO is the largest public dataset available containing instance masks and human pose keypoints.)

- OCHuman is designed for all three most important tasks related to humans: detection, pose estimation and instance segmentation. It is the most challenging benchmark because of its heavy occlusion.

OCHuman v/s COCOPersons

	COCOPersons (val+test)	OCHuman (val+test)
#images	64115	4731
#persons	273469	8110
#persons (oc 0.5)	2619(<1.0%)	8110(100%)
#persons (oc 0.75)	214(<0.1%)	2614(32%)
#average MaxIoU	0.08	0.67
Note : “persons (oc X)” = occluded persons with MaxIoU > X		

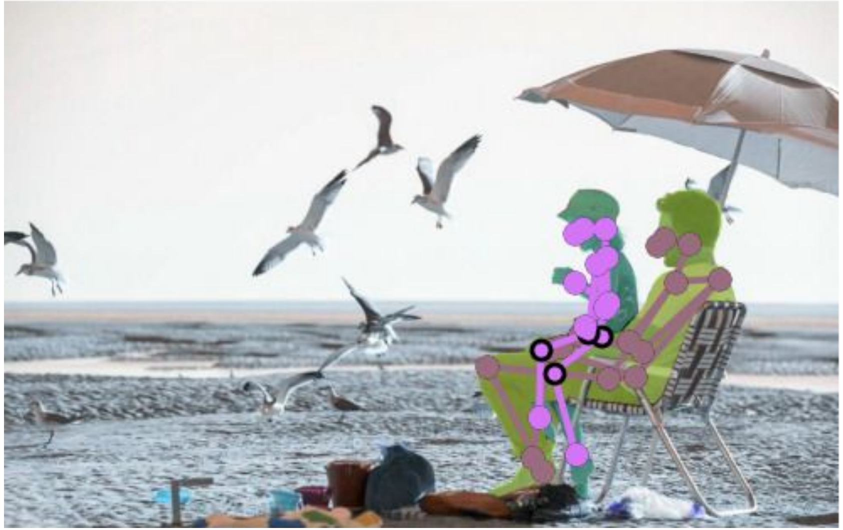
A comparison of the COCOPersons dataset and OCHuman dataset which are the publicly available datasets related to occluded human.

The OCHuman is very challenging from the given statistics.

Section 3 : Results

Inputs : Image and annotations

Input image



Input annotation

Only the pose representation is given as the input for prediction

Outputs of Resnet50 Backbone

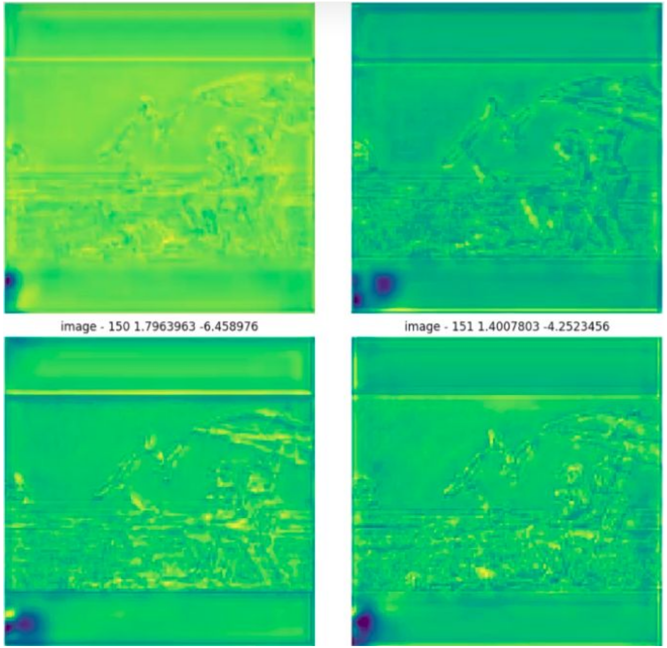
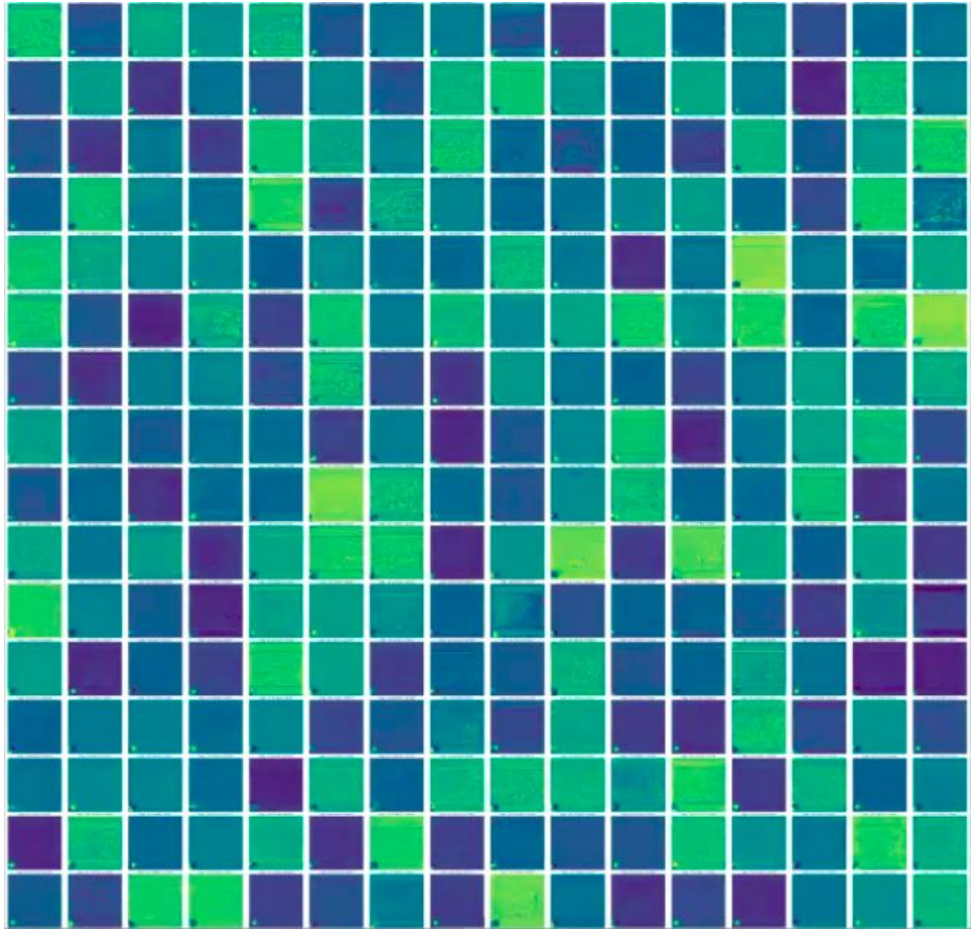
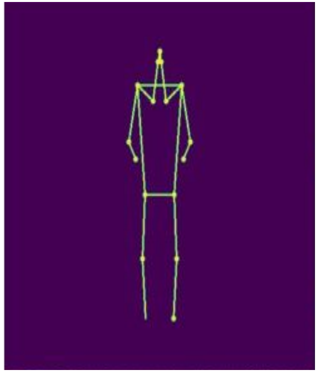


image - 150 1.7963963 -6.458976

image - 151 1.4007803 -4.2523456

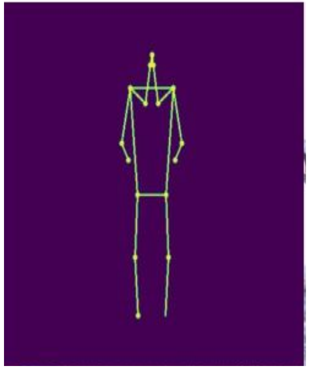


Nearest pose template using Affine Align module



Person-1's Corresponding nearest pose template And H matrix obtained

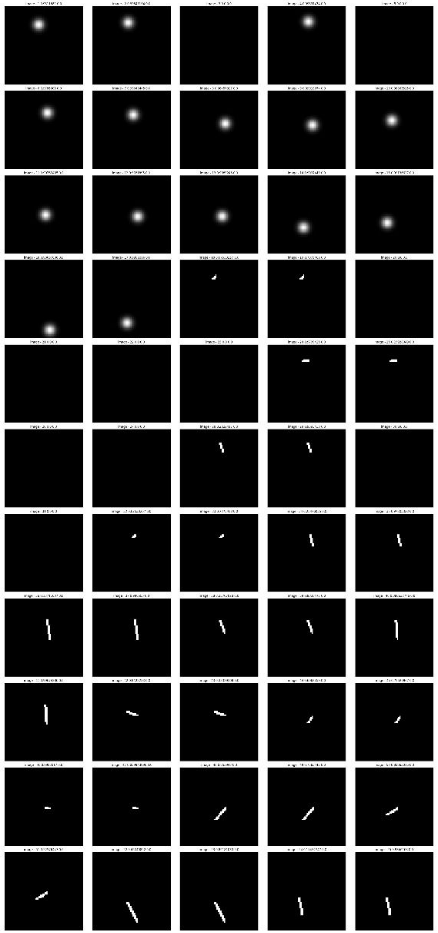
0.35672075	0.29739666	-251.32874
-0.29739666	0.35672075	190.40727



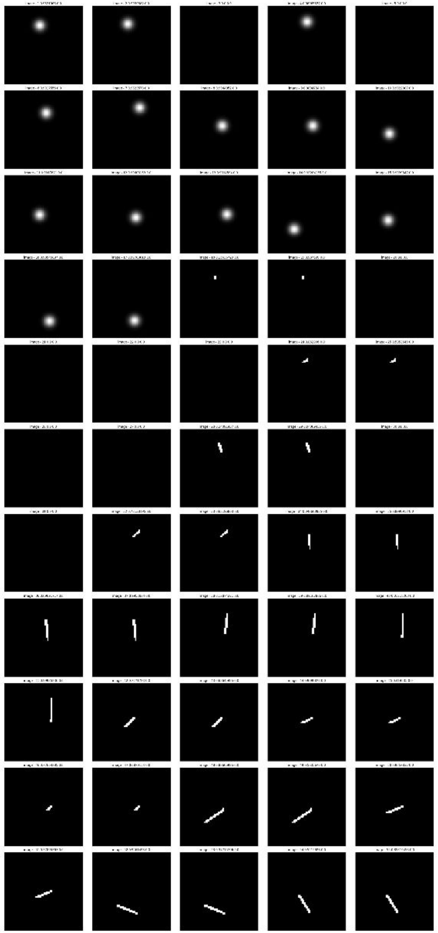
Person-2's Corresponding nearest pose template and H matrix obtained

6.43E-01	1.31E-01	-3.65E+02
-1.31E-01	6.43E-01	-3.14E+01

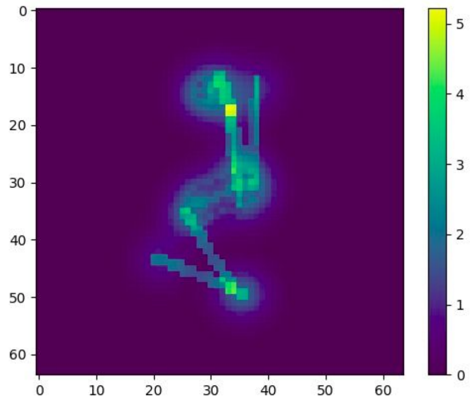
Person 1 :
55 skeleton features



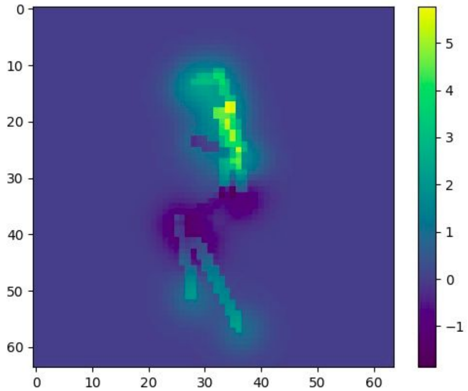
Person 2 :
55 skeleton features



Skeleton features

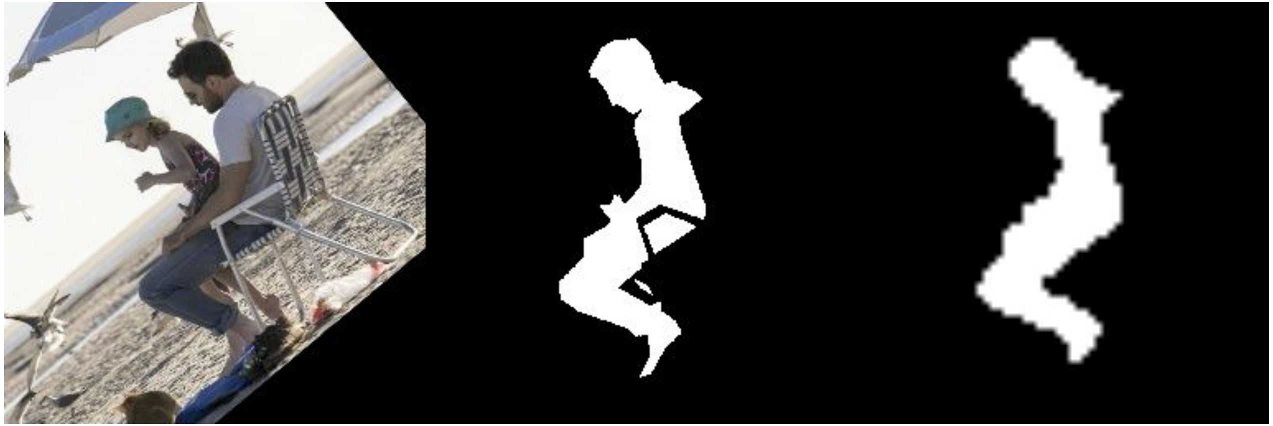


Person 1 skeleton features combined

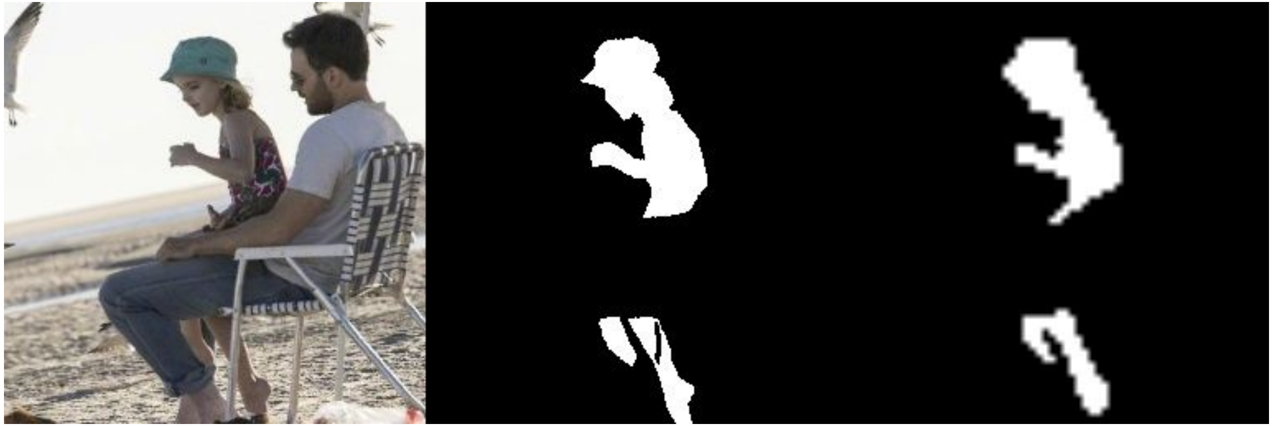


Person 2 skeleton features combined

SegModule output

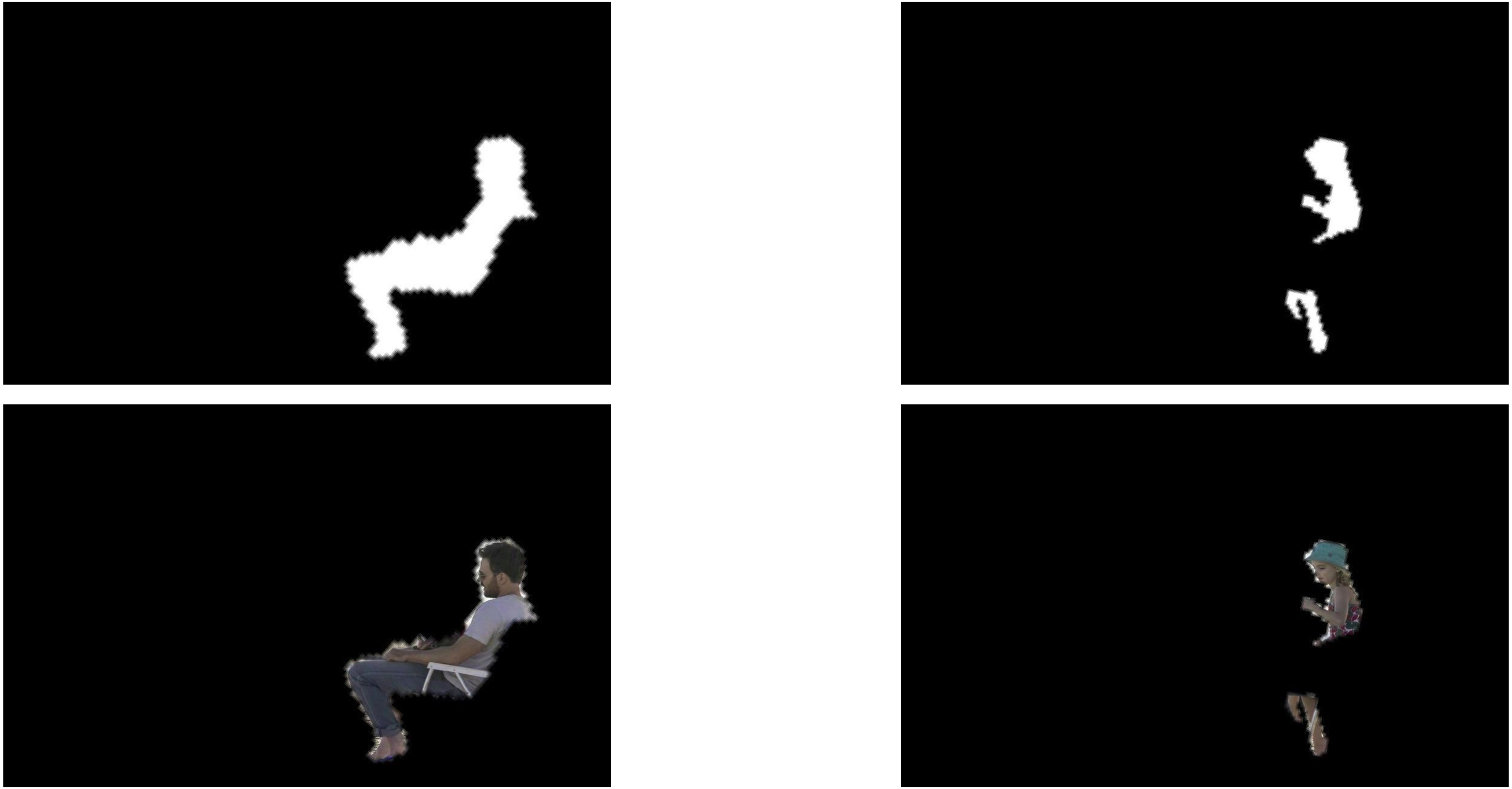


Person 1 segmentation



Person 2 segmentation

Outputs of Align reverse Module



References

Pose2Seg: Detection Free Human Instance Segmentation
Link for the paper : https://arxiv.org/abs/1803.10683

Realtime multi-person 2d pose estimation using part affinity Fields.
Link for the paper :https://arxiv.org/abs/1611.08050

A Blog explaining the paper :
https://towardsdatascience.com/detection-free-human-instance-segmentation-using-pose2seg-and-pytorch-72f48dc4d23e