Extracting Kinematic Information Using Pose Estimation

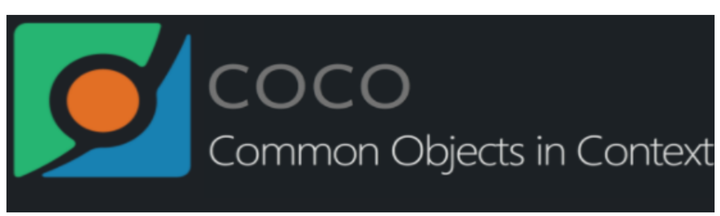
Robbie M. Jones
Department of Computer Science, Stanford University

Motivation

Kinematic information, such as joint angles and joint velocities, are of great importance to fields interested in human motion. For example, athletes may want to have their jumping or throwing motions examined in order to assess injury risk or increase performance. One possibility of recovering this information is through the use of **inverse kinematics**. The inverse kinematics problem takes as input the position of an object and seeks to compute the necessary joint angles and movements to assume that position. Therefore, what we desire is a system to take an image of a person and return coordinates of the person's joints. This is primarily the problem of **pose estimation**.

Dataset

The model is trained on **Microsoft COCO** [2], which is a publicly available dataset for image recognition, segmentation, and captioning. It is widely used in the field of pose estimation because it has **keypoints** for 100,000 people, which are used as ground truth labels for detecting body parts.



Method

We started with **Cao *et. al.***'s [1] network that achieved the highest average precision in the COCO 2016 Keypoint Challenge. The model uses two branches, one to predict **confidence maps** for each body part and another to predict **part affinity fields**. The network is quite large, so we seek to reduce its size and prediction time without substantially decreasing accuracy.

Stage Reduction

One way to decrease the size and runtime of the model is to reduce the number of stages in the latter part of the network. To measure accuracy, we use the outputs of the original 6-stage model as ground truth and compute the L2 loss of the smaller network outputs (Frobenius norm of the difference).

Stage	Scale 0	Scale 1	Scale 2	Scale 3
6 (original)	26.63 ms	71.84 ms	138.20 ms	241.70 ms
5	22.61 ms	61.21 ms	118.64 ms	207.04 ms
4	18.84 ms	50.91 ms	99.05 ms	173.43 ms
3	14.81 ms	40.50 ms	79.10 ms	138.10 ms
2	11.07 ms	30.08 ms	59.28 ms	102.82 ms

Stage	Avg. Heat Map Loss	Avg. PAF Loss
6 (original)	0.0	0.0
5	4.638154	7.762373
4	6.540111	9.555933
3	8.902265	12.255461
2	12.613953	17.647414

Results & Discussion

The displayed images are pose estimations from a reduced model with only 3 stages in the latter part of the network. The smaller network requires half as many parameters for the latter stages, runs ~ twice as fast, and produces visibly identical outputs for the provided images compared to the original, larger model.

Future Work

1. **Smaller Initialization Networks**: The first stage of the network is initialized with VGG-19. Can we use smaller networks (e.g., AlexNet, SqueezeNet) and achieve similar results?
2. **Other Methods:** Network distillation and other forms of compression could prove useful in reducing the network size without sacrificing accuracy.

References

[1] Z. Cao, T. Simon, S. Wei, and Y. Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. *CoRR*, abs/1611.08050, 2016.
[2] T. Lin, M. Maire, S. J. Belongie, L. D. Bourdev, R. B. Girshick, J. Hays, P. Perona, D. Ramanan, P. Dollar, and C. L. Zitnick. Microsoft COCO: common objects in context. CoRR, abs/1405.0312, 2014.