Real time 2D Pose Estimation through PAF

Sriram Kalluri[1] and Sharad Sridhar[2]

[1]skalluri@cs.stonybrook.edu
[2]shssridhar@cs.stonybrook.edu

Abstract

Our project is mainly focused on the 2D pose estimation of multiple people in an image and coming up with an intelligent system to mimic the hand postures. We used the approach presented in [1] for pose estimation which uses Part Affinity Fields to learn to associate body parts in an image. We have enhanced the pose estimation by enabling a tracker mechanism to give an estimate of key points with the help of particle filter and Kalman filter. We have integrated the entire system with Intel Galileo board so that it can mimic the hand postures on servo motors and displays the number of people in an image on an LCD.

Introduction

Estimating the pose of multiple people in images presents a unique set of challenges i.e unknown number of people can occur at any position, interactions between people induce occlusion and limb articulations making the localization of anatomical key points or parts more difficult and run time complexity tends to grow with the number of people in image. Two types of approaches i.e top down approaches and bottom up approaches were used earlier to estimate the pose and both has it pros and cons. We used the bottom up representation of association scores via Part Affinity Fields (PAFs), a set of 2D vector fields that encode the location and orientation of limbs over the image domain. Simultaneous detection and association of key points with the greedy parse is used to achieve high quality results. It may so happen that some of the key points might not be detected properly for certain time or might be missed from the video frame and there is need to enhance the pose estimation.So, we have added an extra feature in the pose estimation by enabling tracking mechanism for certain key points using the particle filter and Kalman filter. The key idea of particle filters is to represent and maintain the posteriori density function by a set of random samples with associated weights and to compute the state estimate from those samples and those weights. Tracking boosted performance of our model by estimating the key points based on the available prior information. Surveillance is an important application in which pose estimation feature can be used. In addition to this, posture recognition is used significantly in remote applications. So, we have come up with an intelligent system by addressing both the features by displaying the count of people in an image on an LCD screen and mimicking the human hand postures on servo motors using Intel Galileo board through Arduino serial interface.

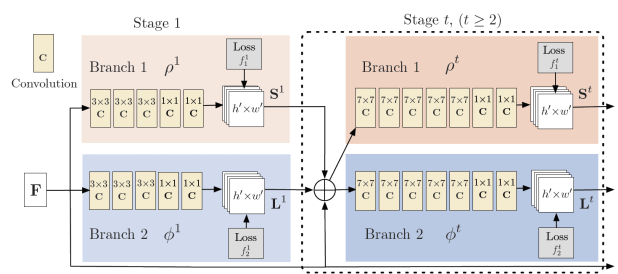

Figure 1: CNN structure

Method

The architecture used in pose estimation simultaneously predicts detection confidence maps and affinity fields that encode part to part association.In figure 1, top branch predicts the confidence maps, and the bottom branch predicts the affinity fields. Each confidence map is a 2D representation of the belief that a particular body part occurs at each pixel location. A single peak should exist in the confidence map if the corresponding part is visible. For multiple people, the confidence map is an aggregation of individual confidence maps via a max operator and the reason behind this is that the precision of close by peaks remains distinct.

Coming to association of parts, one possible way to measure the association is to detect an additional midpoint between each pair of parts on a limb and check for its incidence between candidate part detections. It will rise false associations as it encodes only the position not the orientation of each limb. So to address these limitations, PAF is introduced that preserves both location and orientation in-

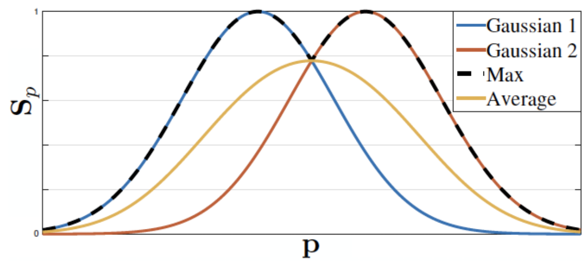

Figure 2: Maximum of confidence map

formation across the region of support of limb i.e the PAF vector field, at an image point p is a unit vector in the direction of limb if p is an point on the limb, otherwise it is 0.

$$L^*_{c,k}(p) = \begin{cases} v \text{ if } p \text{ on limb } c, k \\ 0 \text{ otherwise} \end{cases} \tag{1}$$

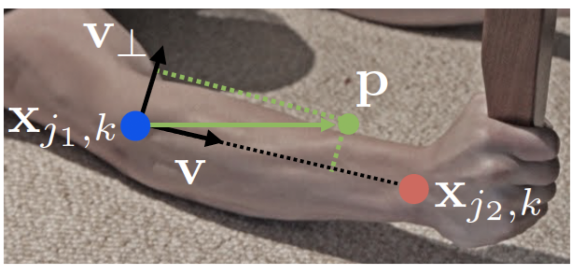

Figure 3: Location and orientation encoding in PAF

Confidence maps are plotted for a sample image for right elbow and the PAFs are plotted for right elbow and wrist. By combining these two, association of the body parts to a respective person is achieved and finally, we can combine those associated points to get the final pose estimation.

Kalman filter is a kind of state observer algorithm designed for stochastic system and is used when the variables of interest can't be measured directly but an indirect measurement is available and is used for linear systems. Kalman gain is calculated such that it minimizes the posterior error covariance. Kalman filter as well as particle filter uses Bayesian inference and gives an estimate based on the prior estimate.

Implementation

We have implemented this entire system in tensor flow using the library provided in [1]. It is properly working for the real time estimation of the pose. One of the issues it may arise here is some of the key points may vanish from the frame for certain instants. To address this situation, we have added the

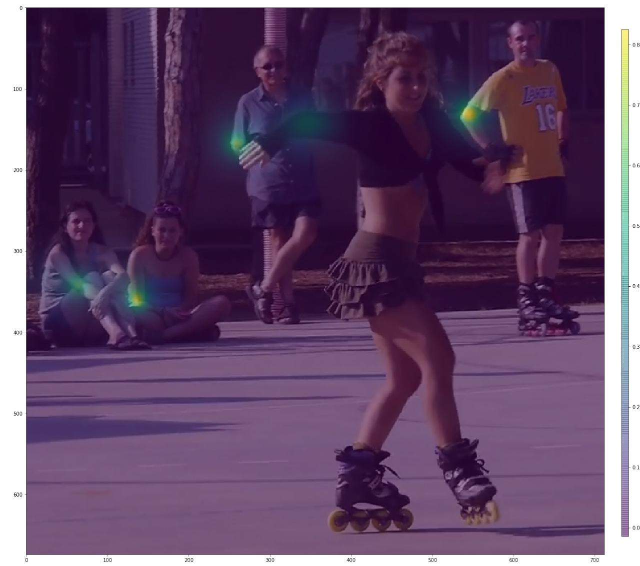

Figure 4: Confidence map for Right elbow

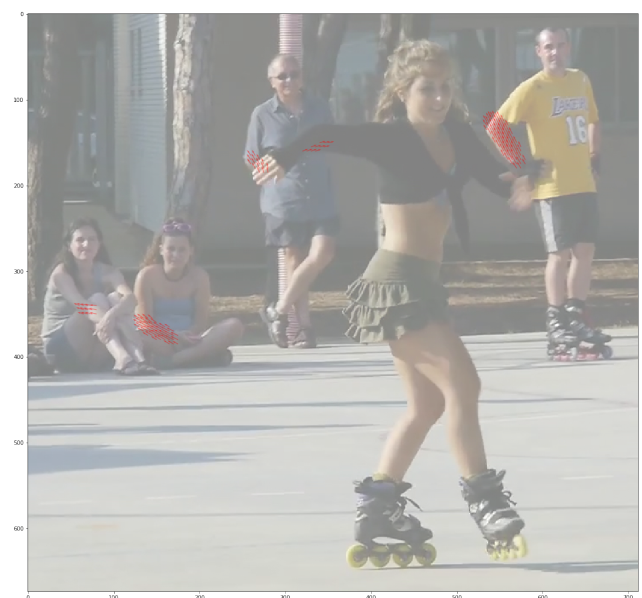

Figure 5: PAF for Right elbow and Right wrist

particle filter which uses the prior information for the estimation of some key points. We have implemented an intelligent system which has 3 modes of operation. First, surveillance mode is implemented to get the real time count of the people in an image and displayed on the 16 X 2 RGB LCD unit which is connected to I2C port on Intel Galileo board. It can be widely used in applications where it is intended to keep the track of the people moving in and leaving by. Second, hand posture recognition and mimicking the postures on servo motors using the measured angles. It has two sub

Figure 6: Pose estimation

modes of operation in which it can mimic the angle between shoulder and elbow as well as the angle between the elbow and the wrist for both the hands using serial communication interface. During our testing, we faced an issue with the angle estimation as it is suddenly changing the angle from -180 to 180. In order to measure the angles for two different modes of operation, we calibrated the angle with certain offset based on the co-ordinates of key points such that it is intended to measure the needed angle for hand posture recognition. This kind of hand posture recognition is important as machines are very accurate at executing those actions by mimicking the postures and it can be highly used in remote applications. Since we are using only 4 servo motors, we have come up with the hand posture recognition. But this can be extended to detecting all the postures and mimicking them in a humanoid robot. So, the interplay between them is highly used in remote applications where human beings are not available directly to perform the task. We connected the Intel Galileo board to COM port and used pyserial library for communication. We have implemented hand shake communication protocol so that it verifies whether the slave received the information from master by sending back the acknowledgement signal to master again. We have implemented the PID part by transferring the angular velocity through pyserial. It is very smooth, but it is very slow as the changes in the real time are fast. Since the motion is quick and to make our system robust with respect to changes observed, we are communicating angle to the servo and the mimicking system is behaving pretty decently. Third, pose estimation is boosted with the help of the tracker mechanism in which particle filter and Kalman filter is used to estimate the position of the key point based on prior available information of those key points.We have written the code for Kalman filter and Particle filter implementation and the integration of that code with pose estimation library. Kalman filter is an optimal solution to many tracking and data prediction tasks and it is very efficient in estimating the position and other relevant information.

Observation and results

Please find the results for the particle filter in figure 7 and kalman filter in figure 8 tracking of face along with pose estimation.

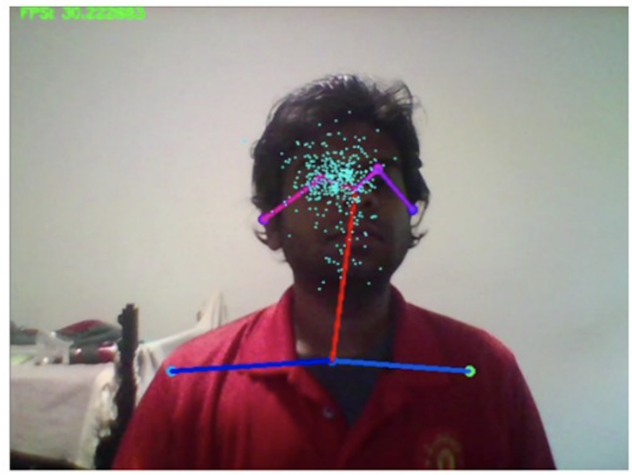

Figure 7: Particle filter

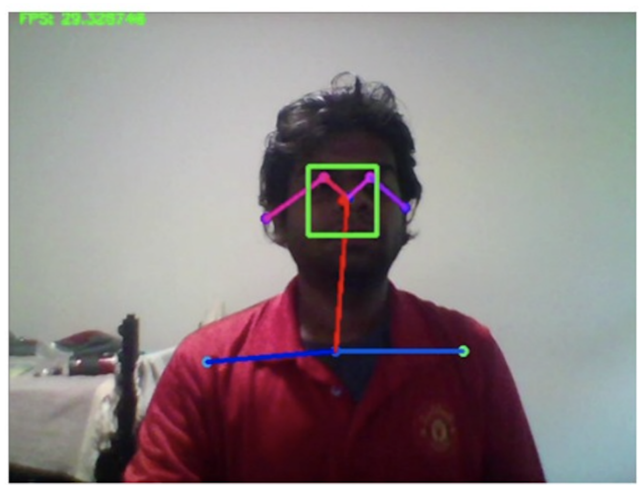

Figure 8: Kalman filter

The results for the pose estimation is shown in figure 9 and mimicking the hand postures in figure 10 on servo board are provided.

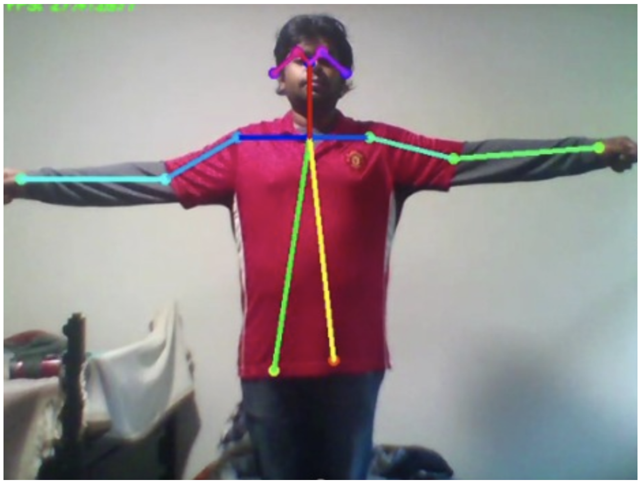

Figure 9: Pose estimation

Figure 10: Hand posture mimicking on servo motors

Here you can observe that our implemented system is accurately detecting the necessary key points for hand posture recognition and we started communication of this data with Intel Galileo board by properly measuring the angles and we can observe that the servo motors are close to 90 degrees when the hand is making an angle of 90 degrees. In our video demo presentation, we have used high K value to show the jerkiness for one servo motor and the optimal K value for other. Real time 2D pose estimation of multiple people is achieved and it can print the number of persons in an image on an LCD screen which can be used for surveillance applications.

Measuring the performance with Kalman filter

In case, one of wrist is not in the frame, then the angle vanishes and the servo is at arbitary value and this will lead to a failure in pose estimation. So, to increase the robustness of the pose estimation, we added the tracking mechanism with the kalman filter to estimate the midpoint of right wrist and right elbow. By using this mechanism, we were able to estimate the midpoint between wrist and elbow even when the wrist is not in frame and servo motor pose detection is implemented. The video in the below link captures both the pose estimation with tracking and mimicking the hand posture in the servo motor when the right wrist is out of the image frame.

https://drive.google.com/file/d/1kaJwNKcdDBZl$_{l}ZCEfX9oqwXAXf_{y}$ $sharing$

Measuring the accuracy of the angle measured

To estimate the accuracy of the angle measured, we have measured the angle for 2 hand postures at 0 degrees and 90 degrees and then 2 more hand postures approximately at 30 degrees and 60 degrees, estimated angles are 1,88,27,58 degrees respectively. The angles measured is based on the key points detected for a particular instant. The key point is one point in the region of the body part. So, inaccuracies are expected in the measurement of the angle and the accuracy for our system is 95.56%. All the codes are uploaded in the blackboard. All the results and videos can be accessed using the following link:

https://drive.google.com/drive/folders/1t2-8ik4ivNBma8VQPV4Dt6nOwx$_{S}Z4jT?usp = sharing$

Summary

Our system is robust to perform the real time 2D pose estimation of multiple person. It is accurate in estimating the hand postures and communicate with the Intel Galileo board for mimicking the hand posture recognition. In cases where certain key points are not detected or not in frame, tracking mechanism with kalman filter increased the performance of the pose estimation in which it is accurately tracking the keypoint even when it is out of video frame.Results and video demos of all the modes of operation are accessible using the above link.

References

1. Cao, Z., Simon, T., Wei, S.E. and Sheikh, Y., 2017, July. Realtime multi-person 2d pose estimation using part affinity fields. In CVPR (Vol. 1, No. 2, p. 7).
2. https://github.com/ZheC/RealtimeMulti-PersonPoseEstimation
3. https://github.com/ildoonet/tf-pose-estimation