Human Pose Estimation in Surveillance Video

Jeet Jivrajani
SEAS - AU
jeet.ji.btechi15@ahduni.edu.in

Priya Mehta
SEAS - AU
priya.m.btechi15@ahduni.edu.in

Harshil Shah
SEAS - AU
harshil.s.btechi15@ahduni.edu.in

Vishwa Saparia
SEAS - AU
vishwa.s.btechi15@ahduni.edu.in

Hiren Galiyawala
SEAS - AU
hiren.galiyawala@ahduni.edu.in

Abstract- Human pose estimation serves as an important parameter for very specific and sensitive computer vision applications. Human pose estimation has been carried out in different ways with different definitions. Pose estimation and soft biometrics like height of person, color of clothes, are used in identification of person from the surveillance videos. This paper focuses on detecting the basic human poses in a video and then tracking it accordingly. Our approach tries to detect basic human poses which include front, back, side faced pose of a human. Use of Faster RCNN is done for detection purpose and four trackers namely KCF, TLD, CSRT, Boosting have been used and compared for the purpose of tracking.

Keywords— Pose Estimation, Detection, Tracking, Faster RCNN

I. INTRODUCTION

Estimating human body poses from images is a demanding task that has attracted great interest from the computer vision community. Determining automatically the body pose promotes many applications such as human tracking, motion capture, activity recognition, surveillance and surgical workflow analysis, context-based image retrieval. There has been significant progress on pose estimation and increasing interests on pose tracking in recent years. At the same time, the overall algorithm and system complexity increases as well, making the algorithm analysis and comparison more difficult.

Human pose estimation is one of the key problems in computer vision that has been studied for well over 15 years. The reason for its importance is the abundance of applications that can benefit from such a technology. For example, human pose estimation allows for higher level reasoning in the context of human computer interaction and activity recognition. Despite many years of research, however, pose estimation remains a very difficult and still largely unsolved problem. Among the most significant challenges are: (1) variability of human visual appearance in images, (2) variability in lighting conditions, (3) variability in human physique, (4) partial occlusions due to self- articulation and layering of objects in the scene, (5) complexity of human skeletal structure, (6) high dimensionality of the pose. To date, there is no approach that can produce satisfactory results in general, unconstrained settings while dealing with all of the aforementioned challenges

The two major activities performed for a human pose estimation in a video sequences are the detection of an object in the frame and the tracking of the same object across all the frames in the video sequences. Talking about Detection, Object detection deals with detecting instances of semantic objects of a certain class (such as humans, buildings, or cars) in digital images and videos. Object detection research work includes face detection and pedestrian detection. Object detection has applications in many areas of computer vision, including image retrieval and video surveillance. It is used in face recognition, video object co-segmentation, also used in tracking objects, for example tracking a person in a video, ball tracking in DRS in cricket. Every object class has its own special features that helps in classifying the class. Object class detection uses these special features. Methods for object detection by machine learning approaches include Viola–Jones object detection framework based on Haar features, Scale-invariant feature transform (SIFT) and Histogram of oriented gradients (HOG) features Followed by Support Vector Machine (SVM) for Classification.

On the other end deep learning techniques that are able to do end-to-end object detection without specifically defining features, and are typically based on convolutional neural networks (CNN). Typical approaches include Region Proposals (R-CNN, Fast R-CNN, Faster R-CNN), Single Shot Multi Box Detector(SSD) and You Only Look Once (YOLO).

While about Tracking, Tracking is the problem of generating an inference about the motion of an object given a sequence of images. It is establishing point-to-point correspondences in consecutive frames of an image sequence. There are two major components of a visual tracking system: target representation and localization, as well as filtering and data association. Target representation and localization is mostly a bottom-up process. These methods give a variety of tools for identifying the moving object. Locating and tracking the target object successfully is dependent on the algorithm. Typically the computational complexity for these algorithms is low. The following are some common target representation and localization algorithms:

Kernel-based tracking (mean-shift tracking): an iterative localization procedure based on the maximization of a similarity measure (Bhattacharyya coefficient). Contour tracking: detection of object boundary (e.g. active contours or Condensation algorithm). Contour tracking methods iteratively evolve an initial contour initialized from the previous frame to its new position in the current frame.

This approach to contour tracking directly evolves the contour by minimizing the contour energy using gradient descent.

Our paper focuses on the detection and tracking of human pose in a video sequence, with Faster RCNN as the detection algorithm and trackers like KCF, TLD, CSRT and Boosting. Introduction of every term is briefly mentioned below:

Faster RCNN:

Faster RCNN is a network that does object detection. It is faster than its descendants RCNN and Fast RCNN. Faster RCNN works as follows. Faster RCNN is composed of two different networks: the Region Proposal Network which does the proposals, and the Evaluation Network which takes the proposals and evaluates classes box. First it run the image through a CNN to get a Feature Map. Then run the Activation Map through a separate network, called the Region Proposal Network (RPN),that outputs interesting boxes/regions. For the interesting boxes/regions from RPN use several fully connected layer to output class + Bounding Box coordinates. The difference here is that Faster RCNN solved the bottleneck of having to run Selective Search for each image as the first step.

A. Kernelized Correlation Filter (KCF)

Kernelized correlation filter is a novel tracking framework that utilizes properties of circulant matrix to enhance the processing speed. The KCF is a variant of correlation filter. In a KCF tracker, the model of the object being tracked is updated online using a linear ridge regression model. Also the number of computation are reduced using properties of circulant matrices and kernel functions, to an order of O(nlog(n)). Given the initial set of points, a tracker tries to calculate the motion of these points by looking at the direction of change in the next frame. In every consecutive frame, we try to look for the same set of points in the neighbourhood. Once the new positions of these points are identified, we can move the bounding box over the new set of points. Having few hyperparameters ,it uses HoG features(32 channels).

B. Track Learn Detect (TLD)

TLD is a framework designed for long-term tracking of an unknown object in a video stream. The object of interest is defined by a bounding box in a single frame. The components of the framework are characterized as follows: Tracker estimates the object's motion between consecutive frames under the assumption that the frame-to-frame motion is limited and the object is visible. The tracker is likely to fail and never recover if the object moves out of the camera view. Detector treats every frame as independent and performs full scanning of the image to localize all appearances that have been observed and learned in the past. As with any other detector, the detector makes two types of errors: false positives and false negative. Learning observes the performance of both tracker and detector, estimates detector's errors, and generates training examples to avoid these errors in the future. The learning component assumes that both the tracker and the detector can fail. By virtue of the learning, the detector generalizes to more object appearances and discriminates against background.

C. Boosting Tracker

This tracker is based on an online version of AdaBoost — the algorithm that the HAAR cascade based face detector uses internally. This classifier needs to be trained at runtime with positive and negative examples of the object. The initial bounding box supplied by the user (or by another object detection algorithm) is taken as the positive example for the object, and many image patches outside the bounding box are treated as the background. Given a new frame, the classifier is run on every pixel in the neighborhood of the previous location and the score of the classifier is recorded. The new location of the object is the one where the score is maximum. So now we have one more positive example for the classifier. As more frames come in, the classifier is updated with this additional data. Tracking performance is mediocre. It does not reliably know when tracking has failed.

D. CSRT

In the Discriminative Correlation Filter with Channel and Spatial Reliability (DCF-CSR), we use the spatial reliability map for adjusting the filter support to the part of the selected region from the frame for tracking. This ensures enlarging and localization of the selected region and improved tracking of the non-rectangular regions or objects. It uses only 2 standard features (HoGs and Color names). It also operates at a comparatively lower fps (25 fps) but gives higher accuracy for object tracking.

II. Related Work

A lot of quality work has been done in the field of Human pose estimation. So in order to understand the current state of art, reference of several good papers has been done to understand the application.

A. Mask R-CNN

K. He et al. propose a conceptually simple, flexible, and general framework for object instance segmentation. Algorithm efficiently detects objects in an image while simultaneously generating a high-quality segmentation mask for each instance. The method, called Mask R-CNN, extends Faster R-CNN by adding a branch for predicting segmentation masks on each Region of Interest (RoI), in parallel with the existing branch for classification and bounding box regression (ref. figure 1). The mask branch is a small FCN applied to each RoI, predicting a segmentation mask in a pixel-to-pixel manner. Fast R-CNN RoIPool is a standard operation for extracting a small feature map (e.g., 7x7) from each RoI. RoIPool first quantizes a floating-number RoI to the discrete granularity of the feature map, this

quantized RoI is then subdivided into spatial bins which are themselves quantized, and finally feature values covered by each bin are aggregated (usually by max pooling). These quantizations introduce misalignments between the RoI and the extracted features.

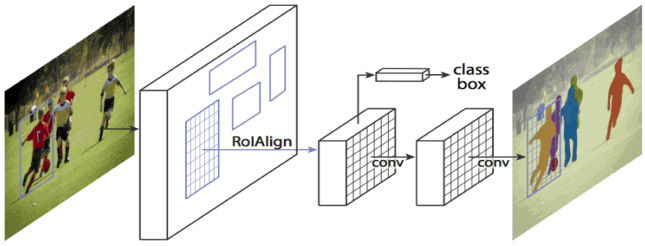


Figure 1. The MaskR-CNN framework for instance segmentation.

While this may not impact classification, which is robust to small translations, it has a large negative effect on predicting pixel-accurate masks. To fix the misalignment, , quantization-free layer, called RoI Align, which faithfully preserves exact spatial locations. Moreover, Mask R-CNN is easy to generalize to other tasks, e.g., allowing us to estimate human poses in the same framework.

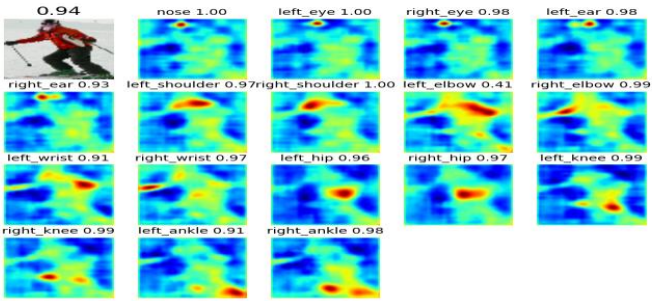
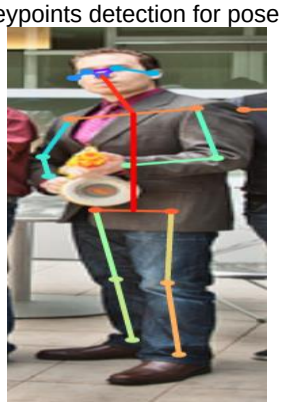


Figure 2. keypoints detection for pose estimation



Mask R-CNN framework can easily be extended to human pose estimation. A keypoint's location is modeled as a one-hot mask, and adopts Mask R-CNN to predict K masks, one for each of K keypoint types (e.g., left shoulder, right elbow ref. Figure 2). This task helps demonstrate the flexibility of Mask R-CNN. For each of the K keypoints of an instance, the training target is a one-hot m x m binary mask where only a single pixel is labeled as foreground. During training, for each visible ground-truth keypoint, the cross-entropy loss over an m^2-way softmax output (which encourages a single point to be detected) is minimized. It was observed that as in instance segmentation, the K keypoints are still treated independently.

Figure 3: Mask R-CNN results

B. Detect and Track

Object detection deals with detecting instances of semantic objects of a certain class (such as humans, buildings, or cars) in digital images and videos. Its research work includes face detection and pedestrian detection.
It has applications in many areas of computer vision, including image retrieval and video surveillance, face recognition, video object co-segmentation and in tracking objects.Every object class has its own special features that helps in classifying the class.

Tracking is the problem of generating an inference about the motion of an object given a sequence of images. It is establishing point-to-point correspondences in consecutive frames of an image sequence.There are two major components of a visual tracking system: target representation and localization, as well as filtering and data association. Target representation and localization is mostly a bottom-up process.These methods give a variety of tools for identifying the moving object. Locating and tracking the target object successfully is dependent on the algorithm.

III. Proposed Approach

The Algorithm followed for the detection and tracking based on the video is shown in fig i.

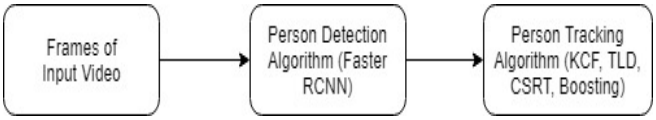
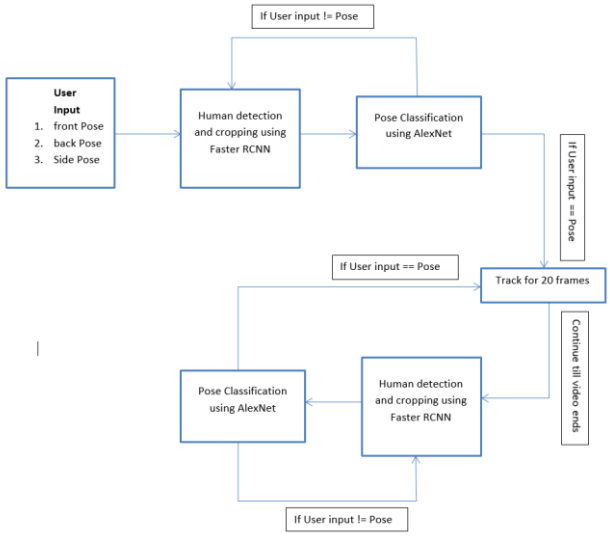


Fig i. Block Diagram

Having got the input as video, we first of all, get all the frames from that video to then perform detection and tracking on frames of the particular video. After getting the frames from the video we individually take the frames one by one and apply Faster RCNN on it for the person detection. Moreover for the tracking purpose we use two tracking algorithms for comparison namely the Kernelized Correlation Filter (KCF) ,Track Learn Detect (TLD), CSRT and Boosting. To reduce time detection is perform after every 20 frame and for the each frame the tracking is continued also focus is kept on reducing the fps.

The pose detection include front, back standing and sitting. The differentiation of the standing and sitting pose is done on the basis of the angle formed by the keypoints in the lower region of the body. Two angles are determined, angle 1 from 11,13,15 and another from 12,14,16. Based on the two angles if the angles are greater than 90, it is standing or else it is labeled sitting.



IV. Experiments And Results

Detection Algorithm	KCF Tracker (FPS)	TLD Tracker (FPS)	CSRT Tracker (FPS)	Boosting Tracker (FPS)
Faster RCNN	24	12	10	22

KCF gave accurate and fast tracking of all the trackers we tested. Frames were adjusted in size as per the person.

In case of TLD, once the bounding box was located on the person, it lost track of the person.

In case of CSRT, as the person moved towards and away from the camera, the bounding box did not adjust according to it. Comparatively lower frames per second.

In case of Boosting, as the person moved towards and away from the camera, the bounding box did not adjust according to it. It had comparatively higher frames per second

The results on the AU surveillance footage.

Tracking / Detection	KCF Tracker (FPS)	TLD Tracker (FPS)	CSRT Tracker (FPS)	Boosting Tracker (FPS)
Only Tracking once detected	78	10	30	24
Tracking and detecting(every 20 frames)	76	6	18	12

The results on mobile video.

Tracking / Detection	KCF Tracker (FPS)	TLD Tracker (FPS)	CSRT Tracker (FPS)	Boosting Tracker (FPS)
Only Tracking once detected	28	12	11	22
Tracking and detecting(every 20 frames)	24	7	8	10

V. Conclusion

The pose estimation done using detection and tracking is on frame basis. First detection is done on one frame and it is matched to the input pose. If it matches tracking is done for 20 consecutive frame and again detection is performed. If it does not match detection is continued till desired results are obtained. Front and back detection is done using face recognition. The siting and standing pose detection is done using the angle formation by the keypoints detected.

Acknowledgment

The authors would like to express their sincere gratitude to Dr. Mehul Raval for his influence on the course of this work and for his suggestions in the improvement of results. We would also like to acknowledge the work done in the state of art which was very helpful during the research.

REFERENCES

[1] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask RCNN. In IEEE International Conference on Computer Vision (ICCV), 2017, Oct 22, pp. 2980 – 2988.

[2] Girdhar R, Gkioxari G, Torresani L, Paluri M, Tran D. Detect-and-Track: Efficient Pose Estimation in Videos. InProceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2018 (pp. 350-359).

[3] High-Speed Tracking with Kernelized Correlation Filters João F. Henriques, Rui Caseiro, Pedro Martins, and Jorge Batista, IEEE TRANSACTIONS ON PATTERN NALYSIS AND MACHINE INTELLIGENCE

[4] Tracking-Learning-Detection,Zdenek Kalal ; Krystian Mikolajczyk;Jiri Matas IEEE Transactions on Pattern Analysis and Machine Intelligence (Volume: 34,Issue: 7 , July 2012)

[5] Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks, Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun; arXiv:1506.01497

[6] YOLO: You Only Look Once: Unified, Real-Time Object Detection,Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi; arXiv:1506.02640v5

[7] Paul Viola, Michael Jones, 'Rapid Object Detection using a Boosted Cascade of Simple Features', IEEE CVPR 2001

[8] Lowe, David G. (2004). "Distinctive Image Features from Scale-Invariant Keypoints". International Journal of Computer Vision. 60 (SIFT)

[9] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 2005, pp. 886-893 vol. 1.(HoG)