Practical Training & Tour Report

On

Human Violence Detection System

Submitted in partial fulfillment of the Requirements for the award of Degree of

BACHELOR OF TECHNOLOGY

In

Computer Science & Engineering

Submitted By

Ankur Goswami (*2017BCSE029*)

Under the Supervision of

Prof. Dr. Ranjeet Kumar Rout

National Institute of Technology, Srinagar
Hazratbal, Kashmir (J & K)-190006

Department of Computer Science & Engineering

2020-21

CERTIFICATE

This is to certify that the Project entitled **Human Violence Detection System** presented by **Ankur Goswami** bearing Registration No. **2017BCSE029** of **Computer Science & Engineering** of **National Institute of Technology, Srinagar** has been completed successfully.

This is in partial fulfillment of the requirements of Bachelor Degree in Computer Science and Engineering, National Institute of Technology, Srinagar, Hazratbal.

I wish her/ him success in all future endeavors.

Prof. Dr. Ranjeet Kumar Rout
(Asst. Prof., Department of Computer Science and Engineering)

Acknowledgements

We would like to take this opportunity to express our sincere gratitude to our mentor, Prof. Dr. Ranjeet Kumar Rout, Department of Computer Science and Engineering, National Institute of Technology Srinagar for his valuable guidance. Not only did his strong insight into the problem domain helped us tackle obstacles, but also invigorated a desire to explore and learn new concepts.

Ankur Goswami

2017BCSE029

Table of Contents

Abstract

With accelerated advancements in surveillance systems, the demand for automated violence detection systems has peaked up significantly. Violence detection in humans has now become a vast & active field of research in Computer Vision. This internship report introduces a method to monitor violence in humans in real-time. The method used, tracks the body orientation of a single person in a frame and analyses sequence of orientations to classify whether the person is violent or not.

1. Introduction

1.1 Why Human Violence Detection System

It is quite obvious that the rate of crimes is increasing day by day in all societies in the world. And the governments are doing everything they could do, spending huge amount of budget to deploy safety surveillance systems & man-power to monitor them & identify the criminals. That’s a great thing to do, but the way it is done, can be more efficient.

The question should be “*Why not Human Violence Detection System?”*.

Thousands of crimes are committed every day, and probably hundreds are occurring right now in the world. To identify them, hundreds of thousands of CCTV cameras are installed in public places & the same amount of workers for each camera at least.

Automating this task using Human Violence Detection System will not only save time but also reduce man-power thus reducing the effective cost.

1.2 Approach & Architecture

This section presents the Human Violence Detection System(HVDS) for detection of violent individuals (persons) in a given or live surveillance video. The approach used in this system is **Human Body Pose Tracking**. The approach is divided into three phases:

a) Human detection & tracking
b) Body keypoints(joints) detection
c) Classification (whether the subject is violent or non-violent)

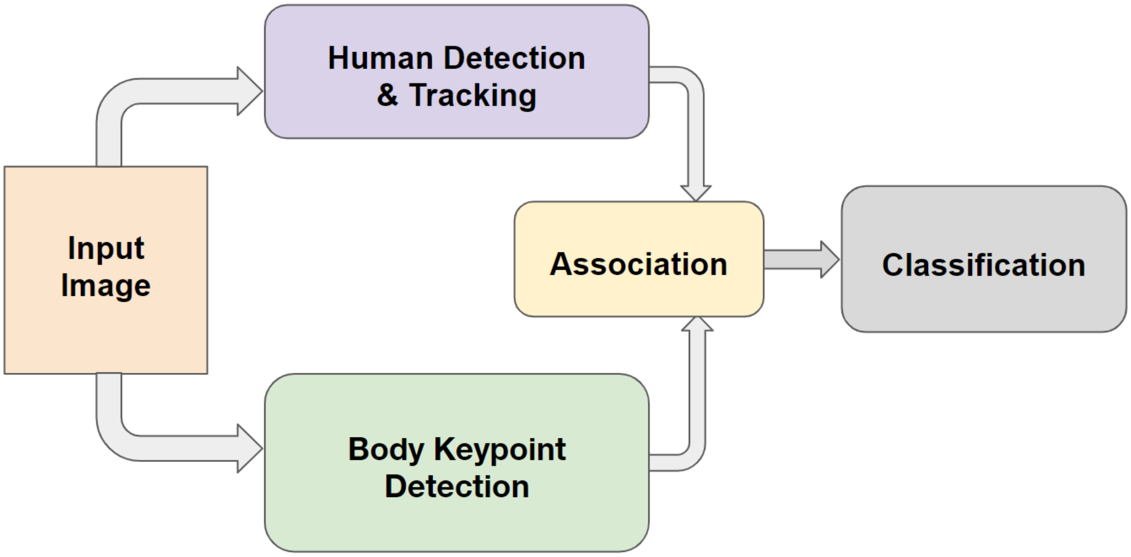

Figure 1.1: General flow chart of Human Body Pose Tracking with three phases.

2. Human Detection & Tracking

In this section, we will discuss Human detection and Tracking. This problem is further divided into two sub problems, i.e.

a) Human detection
b) Human Tracking.

2.1 Human Detection

Since our goal is detecting violence in humans, so we will use a pre-trained deep learning model which can classify humans with high accuracy & speed. The problem of detecting human instances can easily be solved using YOLOv3. To be more specific, YOLOv3-320 can be used because its inference time is 22 ms per frame which is nearly equal to 45 frames per second. YOLOv3 is pre-trained on COCO dataset with 80 classes. We can retrain the network on same dataset with only 1 class, person or not. This will make our model light hence boosting its speed.

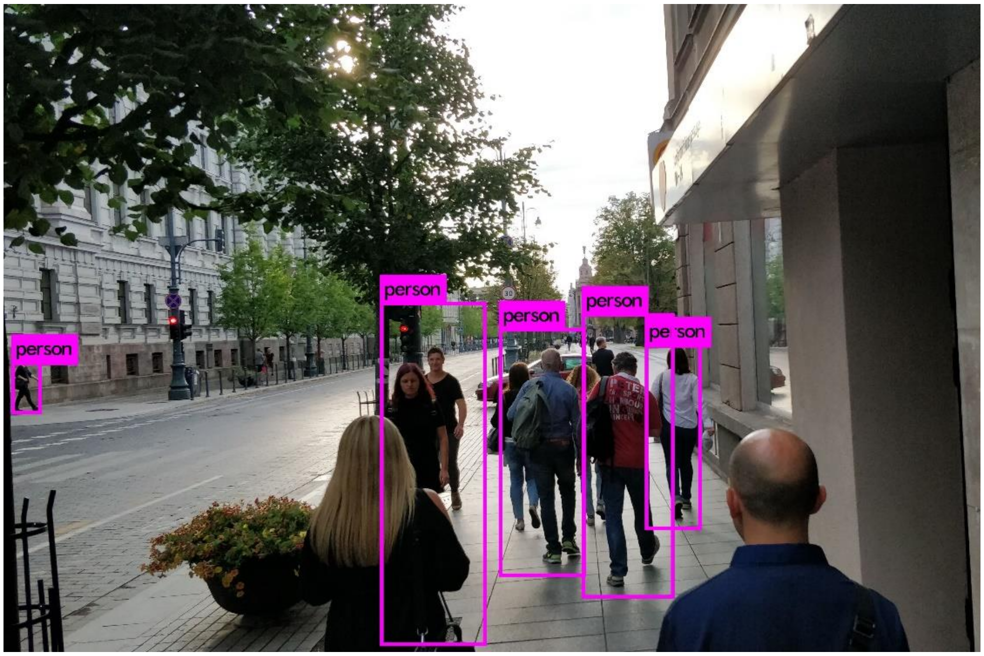

Figure 2.1: Human Detection using YOLOv3

2.2 Human Tracking

As the title suggests, in this section we will track an individual. For that, we use an algorithm called DeepSORT. DeepSORT is a state-of-the-art deep learning model for tracking objects (humans in our case). In this type of tracking, we are expected to lock onto every single object in the frame, uniquely identify each one of them and track all of them until they leave the frame. DeepSORT is immune to common problems one might encounter while using an object tracker such as:

- Occlusions
- Variation in view points
- Non stationary Camera

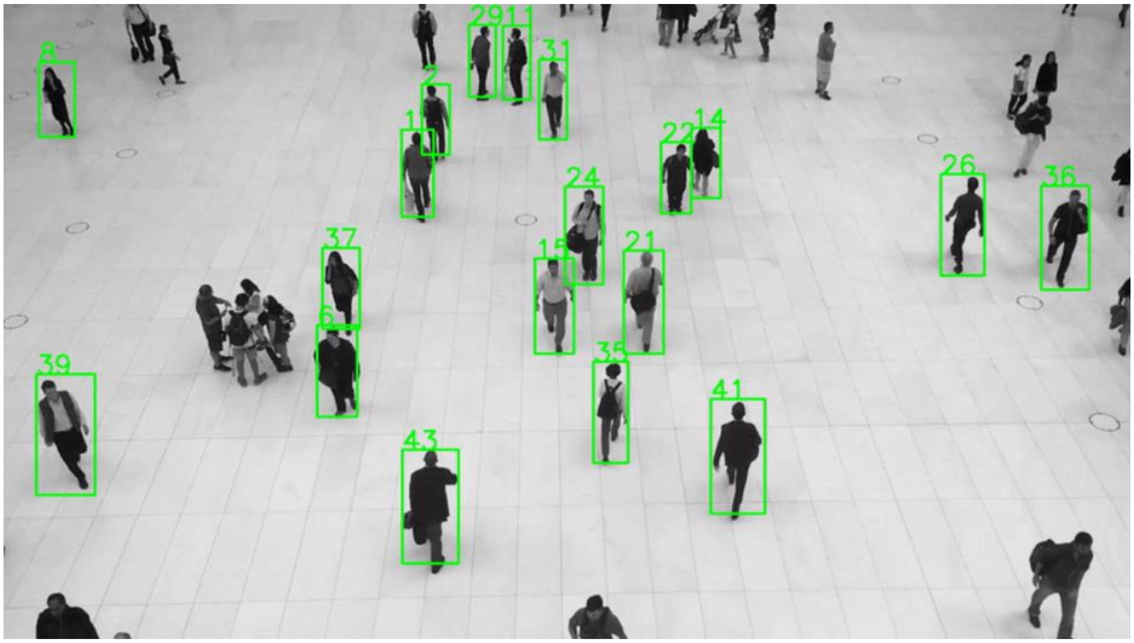

Figure 2.2: Human Tracking using DeepSORT

3. Human Body Keypoint Detection

To detect violence in humans, one must first analyze the body pose of humans. We use the concept of Confidence **Maps** & **Part Affinity Fields** to accurately localize body joints.

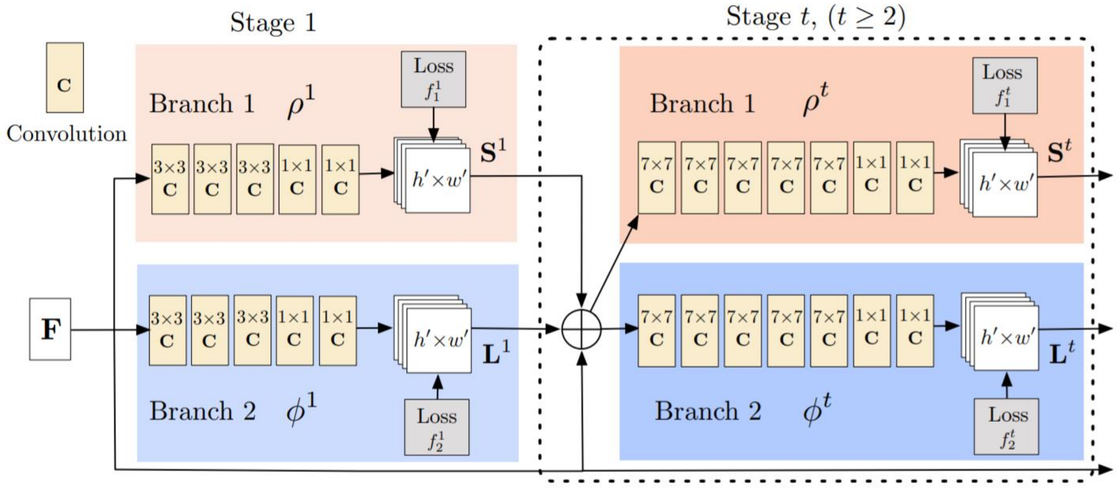

Figure 3.1: Network architecture of body keypoint detection

Confidence Maps for part detection: These are nothing but probability density of a given joint for all persons in a given frame. We first generate individual confidence maps $\boldsymbol{S}_{j,k}^{*}$ for each person k. Let $\boldsymbol{x}_{j,k}$ be the ground truth position of body part j for person k in the image. The value at location p in $\boldsymbol{S}_{j,k}^{*}$ is defined as,

$$\boldsymbol{S}^{*}_{j,k}(\boldsymbol{p}) = \boldsymbol{e}^{-\left(\frac{\|p - x_{j,k}\|_2^2}{\sigma^2}\right)}$$

Part Affinity Fields for Part Association: Given a set of detected body parts, how do we assemble them to form the full-body poses of an unknown number of people? We need a confidence measure of the association for each pair of body part detections, i.e., that they belong to the same person.

We present a novel feature representation called part affinity fields that preserves both location and orientation information across the region of support of the limb. The part affinity is a 2D vector field for each limb for each pixel in the area belonging to a particular limb, a 2D vector encodes the direction that points from one part of the limb to the other. Each type of limb has a corresponding affinity field joining its two associated body parts.

Let $x_{j_1,k}$ and $x_{j_2,k}$ be the groundtruth positions of body parts *j₁* and *j₂* from the limb *c* for person *k* in the image. If a point *p* lies on the limb, the value at $\boldsymbol{L}^{*}_{c,k}(\boldsymbol{p})$ is a unit vector that points from *j₁* to *j₂*; for all other points, the vector is zero-valued. We define the ground truth part affinity vector field, $\boldsymbol{L}^{*}_{c,k}$, at an image point *p* as

$$\boldsymbol{L}^{*}_{c,k}(\boldsymbol{p}) = \begin{cases} \boldsymbol{v} \; if \; \boldsymbol{p} \; on \; limb \; c, k \\ \boldsymbol{0} \; othewise \end{cases}$$

Here,

$$\boldsymbol{v} = \frac{(\boldsymbol{x}_{j_2,k} - \boldsymbol{x}_{j_1,k})}{\left\|\boldsymbol{x}_{j_2,k} - \boldsymbol{x}_{j_1,k}\right\|_2}$$

is the unit vector in the direction of the limb. The set of points on the limb is defined as those within a distance threshold of the line segment, i.e., those points *p* for which $\mathbf{0} \leq \boldsymbol{v}\cdot(\mathbf{p} - \boldsymbol{x}_{j_1,k}) \leq \boldsymbol{l}_{c,k}$ and $|\, \boldsymbol{v}_{\perp} \cdot (\mathbf{p} - \boldsymbol{x}_{j_1,k})| \leq \boldsymbol{\sigma}_{l}$, where the limb width $\boldsymbol{\sigma}_{l}$ is a distance in pixels, the limb length is $\boldsymbol{l}_{c,k} = \left\|\boldsymbol{x}_{j_2,k} - \boldsymbol{x}_{j_1,k}\right\|_2$, and $\boldsymbol{v}_{\perp}$ is a vector perpendicular to $\boldsymbol{v}$.

A pre-trained Openpose's Pose Estimation model can be used to perform the keypoint detection task.

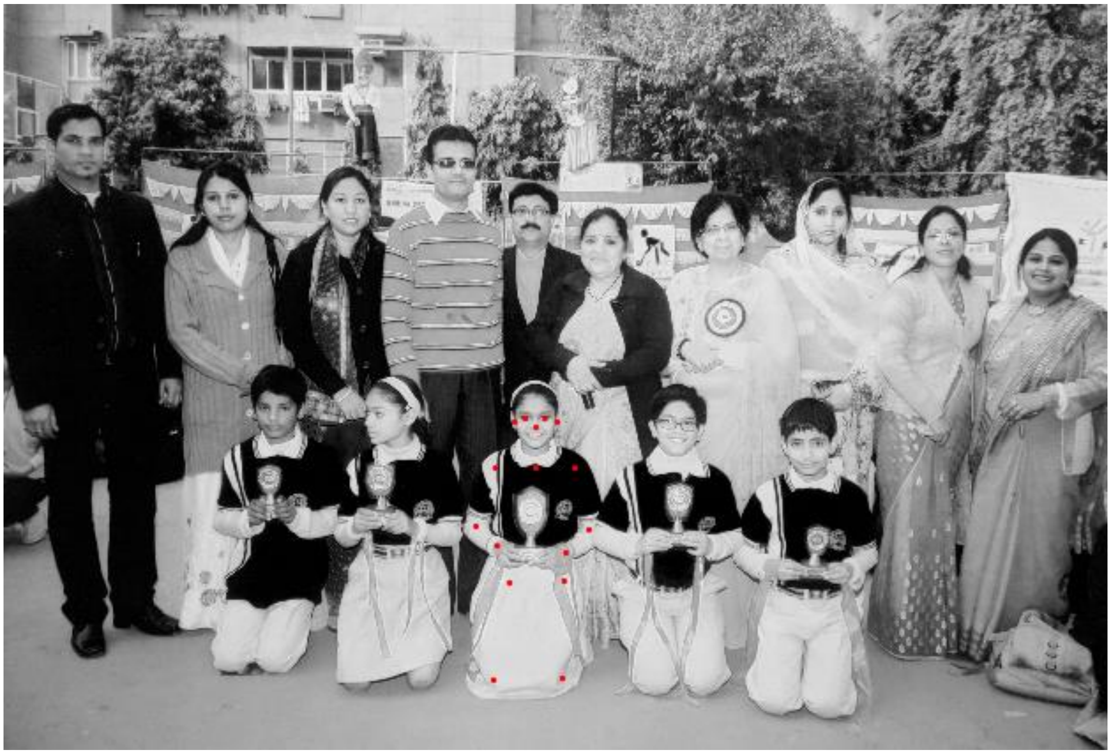

Figure 3.2: Middle person in front row with keypoints.

Association of Bounding boxes with keypoints (Tracking of body keypoints):

Association is done using a brute force method on following criterion:

1. More than 70%* of keypoints must lie inside a given bounding box.
2. The ratio of height of a body (distance b/w top and bottom keypoint coordinate) and height of a given bounding box must be more than 0.7*.

* These numbers are experimental thresholds.

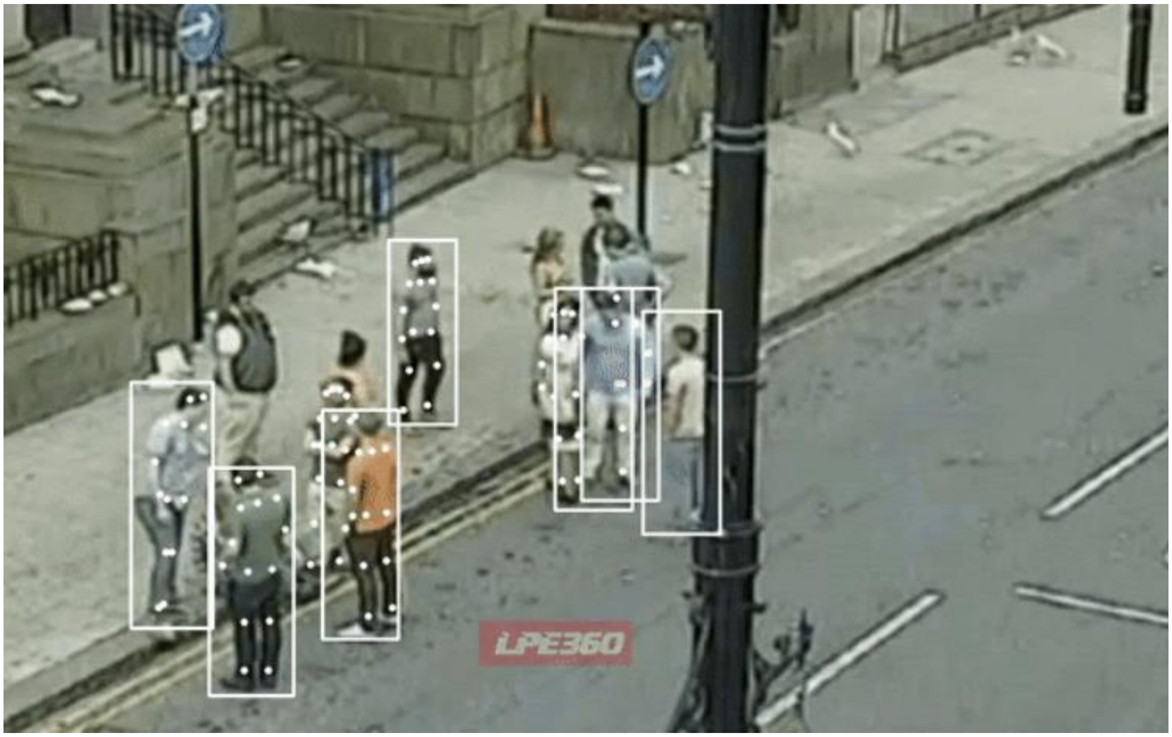

Figure 3.3: Body keypoint tracking

4. Classification

The last phase in human violence detection is classification. Since, spatial arrangement of body keypoints(joints) matters and we are dealing with images, hence classification can be done using Convolutional Neural Networks(CNNs). But there is a problem in using CNNs. CNNs classifies for a frame at any instant. One cannot conclude that a subject is violent or non-violent on the basis of a single image (or a single body pose).

To classify with more accuracy, there should be a method to take past as well as present body movements into account. Recurrent Neural Networks(RNNs) distinguishes themselves from CNNs in this application. Hence, RNNs are better and sensible option for the model.

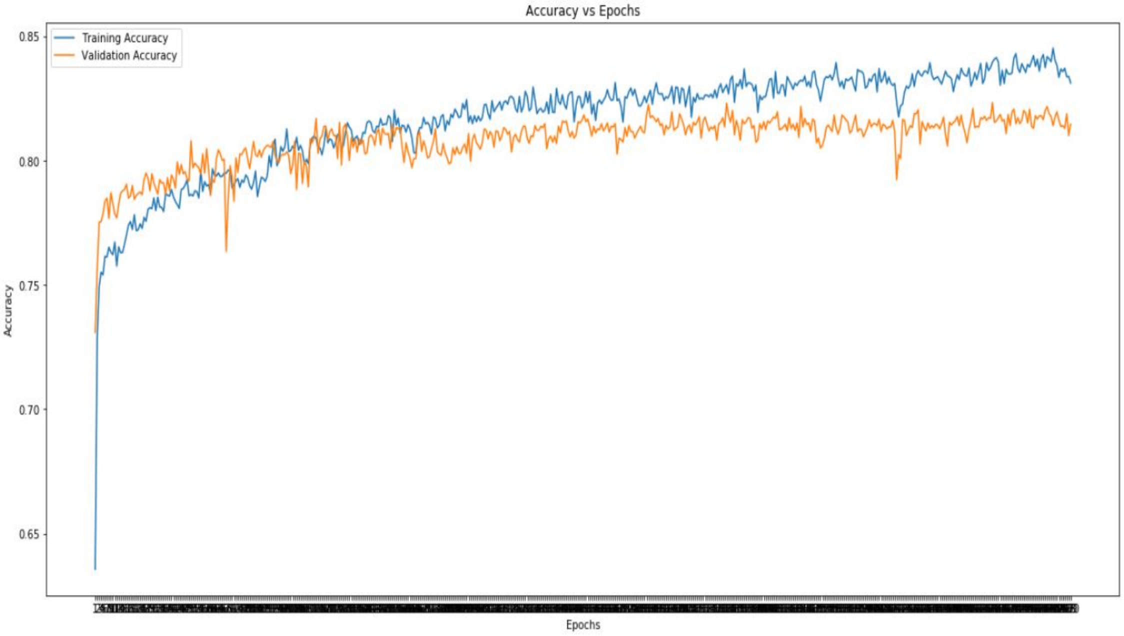

Figure 4.1: Results on 2-Layer deep GRU (Gated Recurrent Unit)

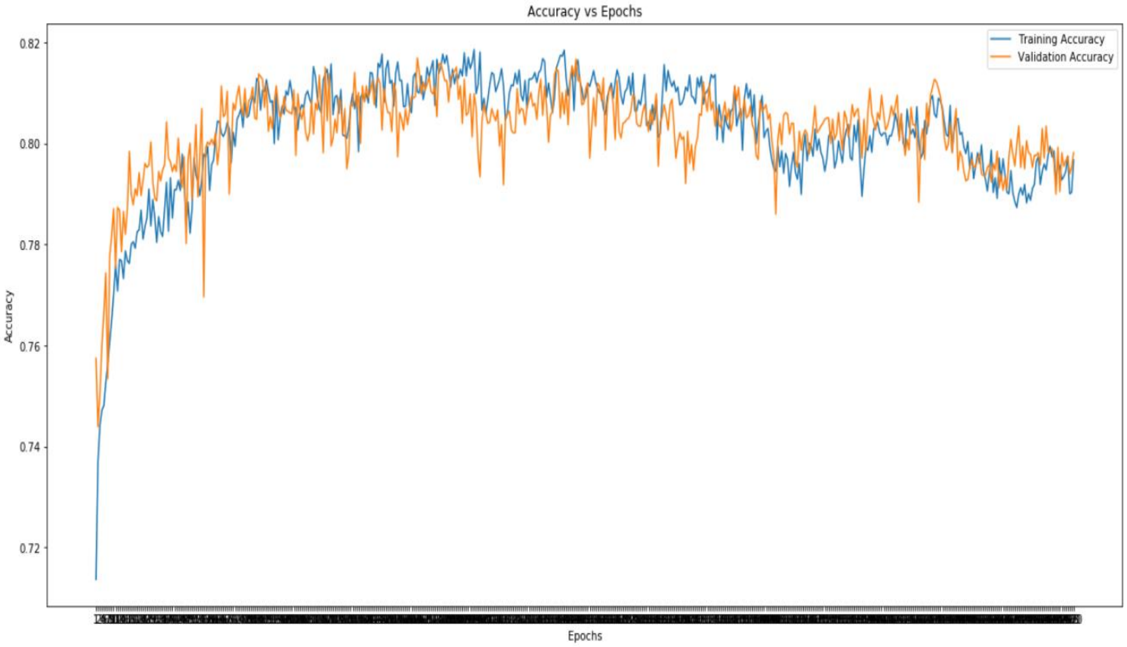

Figure 4.2: Results on 3-Layer deep GRU (Gated Recurrent Unit)

5. Final Model Architecture

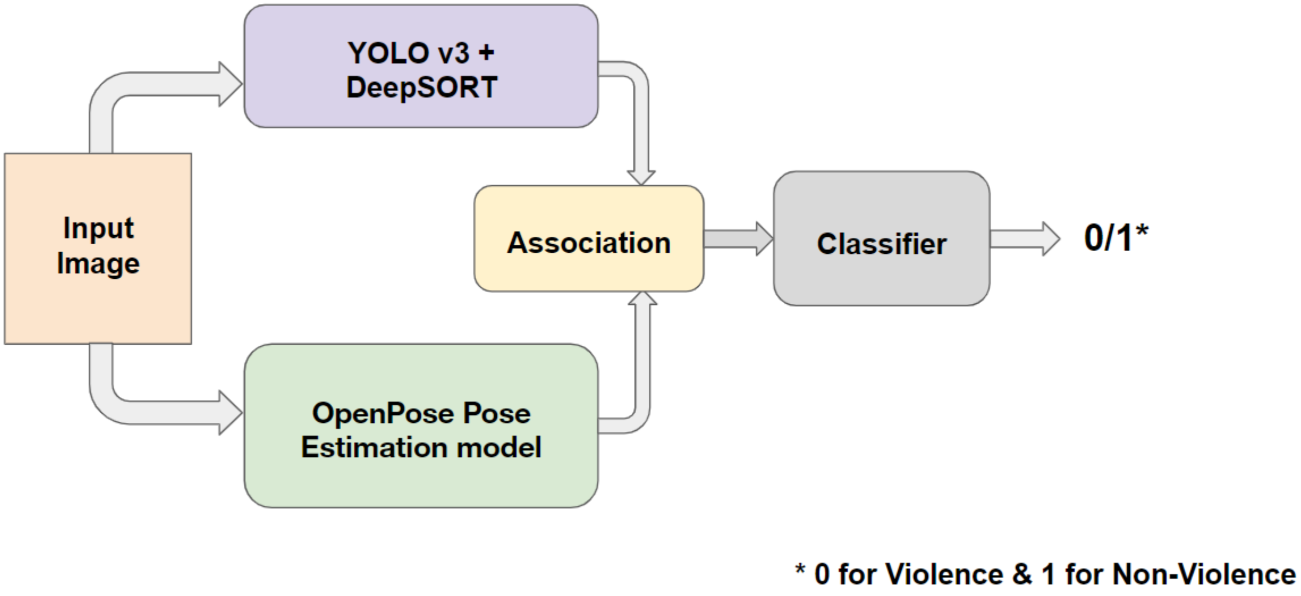

Figure 5.1: Final Model

6. Results

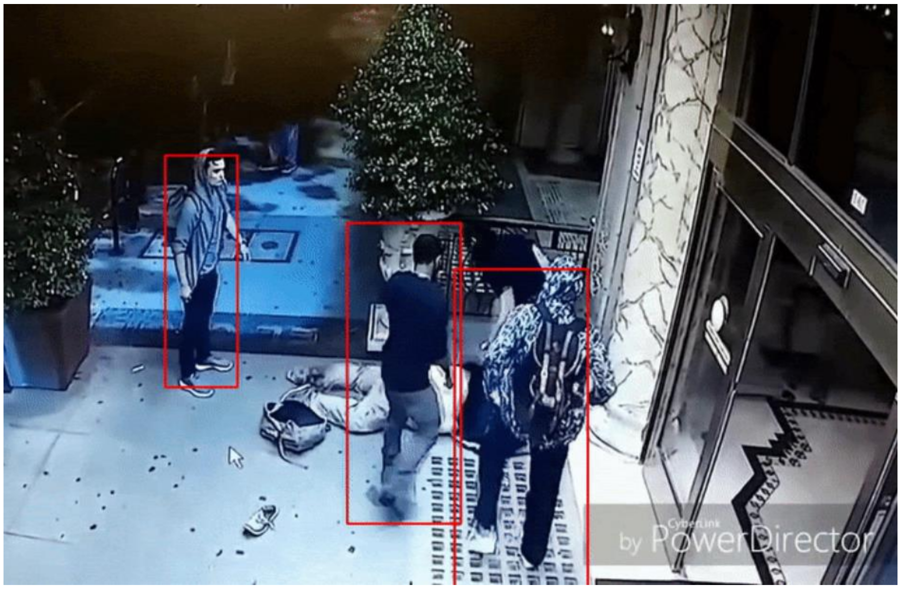

Figure 6.1: Result I

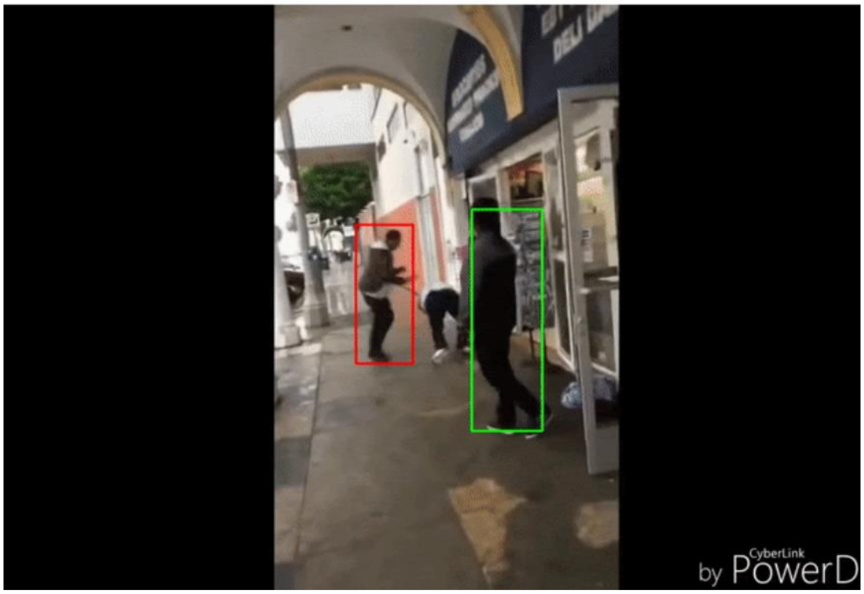

Figure 6.2: Result II

References

[1] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, 2016, pp. 779-788, doi: 10.1109/CVPR.2016.91.

[2] M. I. H. Azhar, F. H. K. Zaman, N. M. Tahir and H. Hashim, "People Tracking System Using DeepSORT," 2020 10th IEEE International Conference on Control System, Computing and Engineering (ICCSCE), Penang, Malaysia, 2020, pp. 137-141, doi: 10.1109/ICCSCE50387.2020.9204956.

[3] Z. Cao, G. Hidalgo, T. Simon, S. -E. Wei and Y. Sheikh, "OpenPose: Realtime Multi-Person 2D Pose Estimation Using Part Affinity Fields," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 1, pp. 172-186, 1 Jan. 2021, doi: 10.1109/TPAMI.2019.2929257.

[4] A. N. Michel, "Recurrent neural networks: overview and perspectives," Proceedings of the 2003 International Symposium on Circuits and Systems, 2003. ISCAS '03., Bangkok, 2003, pp. III-III, doi: 10.1109/ISCAS.2003.1205059.

[5] R. Dey and F. M. Salem, "Gate-variants of Gated Recurrent Unit (GRU) neural networks," 2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS), Boston, MA, 2017, pp. 1597-1600, doi: 10.1109/MWSCAS.2017.8053243.