IAES International Journal of Artificial Intelligence (IJ-AI)
Vol. 10, No. 4, December 2021, pp. 971~981
ISSN: 2252-8938, DOI: 10.11591/ijai.v10.i4.pp971-981

Deep learning techniques for physical abuse detection

Srividya M. S, Anala M. R, Chetan Tayal
Department of Computer Science and Engineering, R. V. College of Engineering, Bengaluru, India

Article Info

Article history:

Received Dec 2, 2020
Revised Jun 10, 2021
Accepted Aug 2, 2021

Keywords:

Convolution neural network
Deep learning
Human action recognition
Human pose estimation
Physical abuse

ABSTRACT

Physical abuse has become a societal problem. Mostly children, women and old age people are vulnerable to it especially in cases of domestic violence or workplace aggression. Reporting it is in itself a challenge especially if there is a pre-existing relationship between the abuser and victim. In this paper we propose a deep learning technique for human action recognition and human pose identification to tackle physical abuse by detecting it in real time. 3D convolution neural network (CNN) architecture is built using 3D convolution feature extractors which extract both temporal and spatial data in the video. With multiple convolution layer and subsampling layer, the input video has been converted into feature vector. Human pose estimation is done using the detection of key points on the body. Using these points and tracking them from one frame to another gives spatial-temporal features to feed into neural network (NN). We present metrics to measure the accuracies of such systems where real time reporting and fault tolerance capabilities are of utmost importance. Weighted metrics shows accuracy of about 89.42% with precision of about 85.82% and thus shows the effectiveness of the system.

Corresponding Author:

Anala M. R
Department of Information Science and Engineering
R. V. College of Engineering
Mysuru road, Bengaluru, Karnataka, India
Email: analamr@rvce.edu.in

1. INTRODUCTION

Physical abuse is any non-accidental act of causing trauma, injury, or other physical suffering or harming body parts. Generally, children are more likely victim to physical abuse caused by parents or siblings or caretakers. Similarly, even old people might be physically abused by younger people. Elder abuse-neglect or mistreatment is often perpetrated by a caretaker, who might be a paid professional or a family member. And those holding higher positions or in power might also physically abuse people in lower cader. The main form of physical abuse is punching, kicking, slapping, beating, caused by fast hand and leg movement. We can leverage emerging technologies like computer vision and deep learning to tackle this problem, specifically detection of child abuse by analysing live video feed from places that are susceptible to physical child abuse such as day-care centres, and nanny cameras. Note that the aspect of real time recognition in this context is of utmost importance and we need to perform both detection, classification and response in real time to possibly alert and intervene when a situation of physical abuse presents itself. Human action recognition and human pose estimation are two of the techniques that are utilized in semantically identifying the actions happening in a video, usually by extracting spatial-temporal features from video frames. The major challenge lies in the fact that an act of physical abuse to a child can be as simple as slapping or kicking them once, meaning the target action might occur only once in a relatively longer video feed. So the system must be robust enough to accurately identify the action and not generate false positives. The goal in this work is to introduce a new way of calculating the accuracy for these spatio-temporal based

data with very few target labels and carry out a comparison study on two different approaches to carry out human action recognition when specifically dealing with physical abuse to another person and come up with a system to make this a viable method of dealing with physical abuse.

2. RESEARCH METHOD

Traditional solutions when it comes to tackling child abuse have generally been geared towards location-based tools which employ the use of mobile devices or tracking devices in order to keep a track of a child's location but require manual intervention from the victim's part to report the incident of child abuse. According to [1] presents a unified platform with an application available on both mobiles as well as desktops that connects directly to Zainab Alert Cell, an operating cell within ZARRA, a federal agency for reducing child kidnapping in Pakistan. This platform combined with GPS powered wristbands, hairbands or other accessories can allow parents to keep an eye on their children along with reporting functionality. The highlight of these proposed bands are fixed coordinates distance pre-set in the band to alert the parents if the child has gone beyond a certain distance and could possibly be in danger. However, the system is location based and requires human intervention to identify the cause of danger.

Location-based solution using geofencing where a parent can define boundaries or 'fences' to monitor the location of children and send emergency messages to local child protection services that are capable of monitoring children activity was proposed in [2]. The solution relies on knowledge of the social environment to determine location of interests and possible harmful locations. Again, this solution relies on devices capable of operating accurately and reliably as a location reading device located with the child. In 2018, Shruthi [3] also provides a solution that is dependent on wearable devices with location tracking combined with alert systems. This requires a possible victim sending a SOS request using their own mobile device. They also propose a higher functioning GPS using antennas located on clothes as logos so as to remain disguised from potential abusers. In 2016, Jatti *et al*. [4] propose a machine learning algorithm to evaluate signals and determine if an individual is relaxed or stressed. If the individual is determined as stressed, temporal based action recognition is proposed to alert nearby authorities. However, looking at the national statistics provided by Health and Human Services Department of the Children's Bureau of the US [5] on child abuse for the year 2018, it is observed that 92% of the children were victimized by a parent. Looking at the perpetrator-victim relationship numbers provided in [6], it was seen that 77.5% of the perpetrators in 2018 were a parent of the victim, 6.4% were a relative other than a parent and 4.2% had a multiple relationship (such as a nanny or caretaker) with the victim. This means 88.1% of the total reported cases were carried out by a person which the victim knew, meaning there is a much lower probability of the child being moved to a location further than their designated safe area.

Hence in recent times, approaches based on deep learning have gained maximum popularity in the research on vision based human action recognition. Complex human to human interaction can be analysed. They have the ability to learn from simple to complex features. By having multiple layers of processing and high-level representation of the given input video can be built. Deep learning approach uses weight sharing, local perception, down-pooling, and a multi-convolution kernel to learn local information from part of an image. The result will be the output from final recognition layer. The final recognition layer will be determined by the result of multiple layers of convolution. And action recognition is [7] considered to be a combination of gestures, for instance, “running”, can be determined as a combination of arm and leg gestures. The various challenges and approaches involved in human action recognition are discussed in detail [8]. Spatiotemporal related specific features are considered in [9] for action. In 2010, Poppe [10] gives a detailed survey on various vision based action recognition systems discussing the challenges of classification algorithms and image representation. Human action analysis [11] techinques were reviewed and approaches on invariable view on pose detection and behaviour were discussed. In 2011, Weinland *et al*. [12] presented a survey on the approaches for action recognition, representation and segmentation in vision-based approaches. Some datasets on human action and activity recognition are reviewed in [13].

2.1. Data set

The most challenging part of this work is creating data for physical abuse. There are no publicly available datasets that come under the umbrella of physical abuse in a child-parent environment and although there are datasets like UCF101 and kinetics action recognition dataset which have examples for actions like boxing and punching, the inputs are of much lower resolution and does not match the description of having long instances of video feed with single action happening, which can make a system trained on data like this highly susceptible to false positives. Given all these factors into consideration, the testing data was generated, by recording video clips.

In order to create action data, video clips were recorded of varying lengths for the following targets: stand, kick and slap. 4 clips for each target were recorded, with 2 clips having one participant and 2 clips having 2 participants each. The actions for our training data were performed in different orientations in each clip to account for as much variance in the positioning of the people as would be present in a real-life video feed. Figure 1 shows different frames from recorded videos with different orientations and number of people. 36 short clips were sampled from each video with each clip being 5 frames.

Figure 1. Dataset with different orientations and number of persons

This equates to 36 short clips from 12 examples each, resulting in 432 samples of 5 frames each and training data of shape 432x5x360x480x1 which matches the format

$$NxDxHxWxC$$

where:
N: Number of Examples,
D: Depth,
H: Height,
W: Width of frame and
C: Number of channels

This is a single view-based dataset. The datasets use a single camera for recording human actions from a certain fixed angle without camera being moved. This dataset has 12 recorded actions and each action was performed by either single or two individuals.

2.2. 3D convolution neural network (CNN)

In 3D CNN architecture, various features are extracted using multiple convolution operations. They are applied at the same location on the input. Then, convolution and subsampling will be performed for the channel from adjacent frames. Information from all channels will be combined to form final feature representation. 3D CNNs have seen an increased growth in the recent years due to their ability to extract features that are both high level and low-level representations from images. A typical convolution operation can be defined as a mathematical function using the given representation:

$$S(i,j) = (K * I)(i,j) = \sum_m \sum_n I\,(i-m, j-n)\,K(m,n)$$

S : feature map
K : kernel of size *m x n*
I : input image of size *i x j*

Kernel *K* is spatially smaller than Input *I* and the resulting convolution operation produces a resulting feature map that is able to capture details about our image with learnable weights associated with *K*.

The dimension of the feature space after applying the filter is governed by the formula

$$\left\{\frac{N+2P-F}{S}\right\}+1 \quad (1)$$

N = input image dimension
P = pooling size
F = filter size
S = stride

While this operation allows us to map features in an image such as edges and lines on to the feature space and perform well on tasks like 2D image classification and object detection/localization, 2D CNNs perform poorly on data with underlying temporal features, such as video classification due to its inability to correlate temporal dependencies making them unsuitable for video action recognition which inherently relies on the system's ability to capture temporal dependencies from one frame to another in a video clip by utilizing a 3-dimensional feature that extracts features along 3 dimensions of width height and time. Instead, we can utilize 3D convolutions that are better at capturing a temporal based feature. Figure 2 shows a 3D convolution operation using 3D filter.

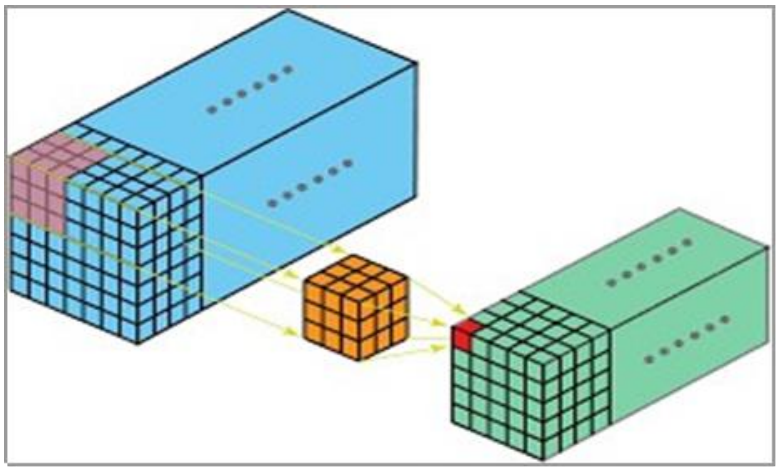


Figure 2. 3D convolution operation by using a 3D filter

Note that the dimensions of the output feature map follow the same rule as (1) for calculating the output height, width and depth. The formula for a 3D convolution operation is as:

$$S(i,j,k) = (K * I)(i,j,k) = \sum\sum\sum I(i-m, j-n, k-p)\, K(m,n,p)$$

3D-CNN is composed of three 3-dimensional conv layers and 2 fully connected layers. The input shape to our first layer is 5x360x480x1. Each output is followed a ReLu non-linearity except the last output layer which has 3 output nodes followed by a SoftMax non-linearity to generate output probabilities for each of our three classes. The last Conv3D layer is also followed by a 3D MaxPooling layer to reduce the receptive field of our feature map. We have also utilized valid padding i.e p=1 throughout the network. The shape of our kernels used is [1x3x3]. Losses are then calculated using the log loss function. We trained our model for 1 epoch and use Adam optimizer to converge the network. While training, our dataset was split into training and validation using 410 examples for training and 22 examples for validation. After training, we were able to achieve a training accuracy of 93% and a validation accuracy of 99%. We will compare our test results for both the techniques in a later section.

2.3. Human action recognition (HAR)

HAR works on classifying the activity being carried out by a human present inside a video. The challenge while tackling this problem lies in the aspect of carrying out a significant number of classifications each second in a video sequence and the spatial-temporal nature of the data itself i.e. sequence of video frames over a period of time. Vision-based HAR has applications in human- computer interaction, health care, video surveillance and many such areas. Typically, global representations have been utilized to represent a video or image and encode it as a feature. Then localization and region of interest is identified.

However, this two-step process is not sufficient for action recognition which is in real-time. Generally, it is difficult to train large 3D CNN on 3-dimensional data that is typically needed to create models capable of carrying out action recognition [14]. Approaches are used [15] of training large 3D convolutional networks in order to classify actions [16] in a video by treating stacked frames as the third dimensional input to the 3D CNN and extract temporal features from one frame to another and perform human action recognition [17].

A 2-way convolutional approach, utilizing both optical flow as well as features extracted from RGB input and combining them to generate a representation of motion in temporal space is presented in [18]. Features are extracted from images using a CNN [19] and then using these feature vector as input to an RNN [20] to classify long term dependencies in a video sequence and coined the architecture as LRCC or long-term recurrent convolutional neural networks. According to [21] was developed at Facebook and currently holds SOTA results for many popular activity recognition datasets like THUMOS, UCF101, Kinetics and HMDB activity recognition datasets. However, a general problem with these approaches is the underlying datasets that are trained on. It doesn't generally translate well into a human-child interaction environment and do not scale well to the input sizes when dealing with input feed from webcams. This makes it terribly slow when trying to carry out activity recognition in real time. However, this is where this work utilizes human pose estimation. This approach is similar to [22] where it utilizes human skeletons generated from human pose estimation and use the skeletons as input to the model. The approach discussed in [23] also offloads most of the heavy computing to the task of generating human pose and [24] uses a simple network architecture to make predictions using human pose features instead of using a big architecture for carrying out human action recognition on the original input itself. This approach makes it very promising for real time systems with very limited fault tolerance. Human pose using CNN is handled in [25] especially for occlusion using detection and regression. Multicontext attention mechanism is used in CNN [26] to gain the ability to focus on different granularity. Learning feature pyramids are designed in deep CNN models to handle scale changes [27].

2.4. Human pose estimation

Human pose estimation is done with localization of the joints of a human in an image or a video and uses it to estimate the pose of the human. It consists of jointly detecting keypoints on the body, hand, face and foot. Typically, there are two approaches for carrying out human pose estimation - top- down and bottom-up approach. Top-Down approach involves localization of humans in the frame using a bounding box detector and then roughly guessing the pose of a single person in that box whereas the bottom-up approach starts by localizing the keypoints in the image and then grouping those keypoints into the instance of a person. Using the location of these joints and the subsequent movement of the joints from one frame to another allows to extract these joints as spatial-temporal features to feed into the neural network.

OpenPose [22], is a library developed by CMU that identifies keypoints in human face, hands, legs and body. It does multi-person keypoint detection for multiple persons on a single image or a video. It was trained on the CMU Panoptic Studio dataset and is capable of carrying out both 2D and 3D real-time multi person keypoint detection and draws a skeleton overlapping the human in the image. This generated skeleton data is used by our motion-estimation algorithm to classify different actions being carried out in the video feed by a human.

The skeleton generated has 18 joints that includes areas in the head, arms, neck and legs and is show in Figure 3. It uses Part Affinity Fields which are a set of 2D vector fields that encode the location and orientation of limbs over the image domain. It uses bottom up approach. These are able to encode unstructured pairwise relationships between body parts of a variable number of people present in the image.

First a feed-forward neural network predicts a set of 2D confidence maps S of body part locations and set of 2D vector fields L of Part Affinity Fields that encode a degree of associativity between different body parts of the skeletons.

$$S = (S1\ , S2\ \dots\ Sj\ )$$

$$L = (\ L1\ , L2, \dots Lc\ )$$

S has j confidence maps with one confidence map per part while L has c vector fields with one vector field per limb. The Figure 4 shows the design of OpenPose's multistage architecture for generating heatmaps of the skeleton poses.

The image is analysed by a CNN (first 10 layers of VGG- 19 as a base) generating feature maps F that acts as input to the first stage of the network architecture. Then in the first stage, one branch predicts a set of confidence maps $S1 = p1(F)$ while the other branch predicts a set of Part Affinity Fields $L1 = o1(F)$ where $p1$ and $o1$ refers to the CNN used for carrying out the inference at Stage 1. In the following stages,

predictions from both of these branches is used along with the orignal image feature by concatenating them and use them to produce much more refined predictions.

$$St = P(F, St-1, Lt-1)$$

$$Lt = O(F, St-1, Lt-1)$$

where *P* and *O* are the CNNs used for carrying out inference in stage *t*.

The motion estimation algorithm is used to extract features from the skeleton data that is then used by a Neural Network for classification. The skeleton data is taken from 5 frames of the video each time and then computes the following features:

-*Xs*= Concatenating joints positions of 5 frames
-*H*=Average height of skeleton of previous 5 frames
-*V_body* =$Velocity\ of\ neck/H$
-*Normalized Joint Positions* =$Xs - mean\ (Xs)/\ H$
-*Velocities of joints* =$X[tk]\ -\ X[tk-1]$
-Joint angles computed from joint positions.
-Length of each limb computed from joint positions.

After experimentation, features 3, 4 and 5 were used and concatenated to form a feature vector of dimension 314. PCA was applied to reduce the dimension of this vector to 50 dimensions. This vector is then used as input to a neural network consisting of 2 hidden layers, each with 100 nodes and the output layer consisting of 3 nodes to serve as our predictions. This network is the default network available in scikit learn library for neural networks.

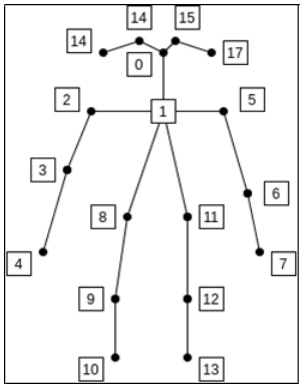


Figure 3. 36 values, each point being represented as (xi, yi), i= 0 to 17

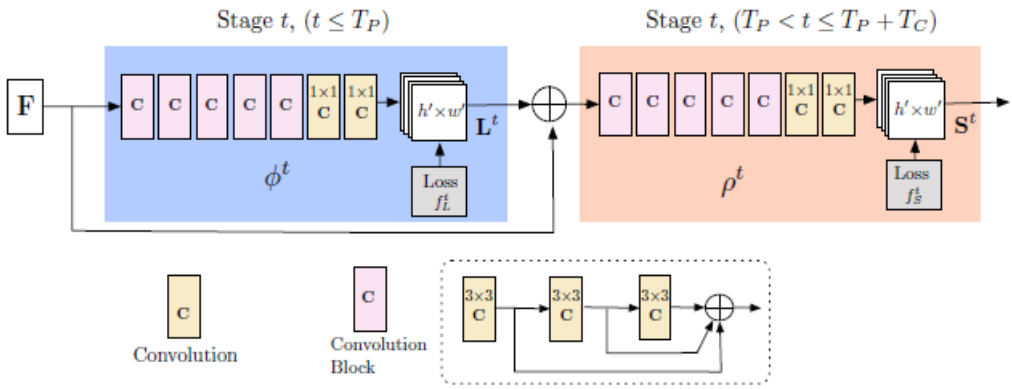


Figure 4. Multistage architecture of openpose [22]

3. RESULTS AND DISCUSSION

There are 3 classes that a video clip can be classified into standing, kicking and slapping. Two categories are also defined that the actions can be split into: GOOD and BAD categories. The actions of kicking and slapping belonging to the category BAD and the action of standing belonging to the category GOOD. The main target for the system is to identify whenever a bad action occurs in a video and to also not incorrectly classify no action happening as a bad action. Based on this, the metrics are dined as in Table 1. Using these target metrics, a novel way of prioritizing the weight of the actual prediction are developed as explained in section 8.

Table 1. Definition of accuracy metrics

Metric	Definition
True Positive	Bad Action - Present Prediction - Bad
True Negative	Bad Action - Absent Prediction - Good
False Positive	Bad Action - Absent Prediction - Bad
False Negative	Bad Action - Absent Prediction - Good

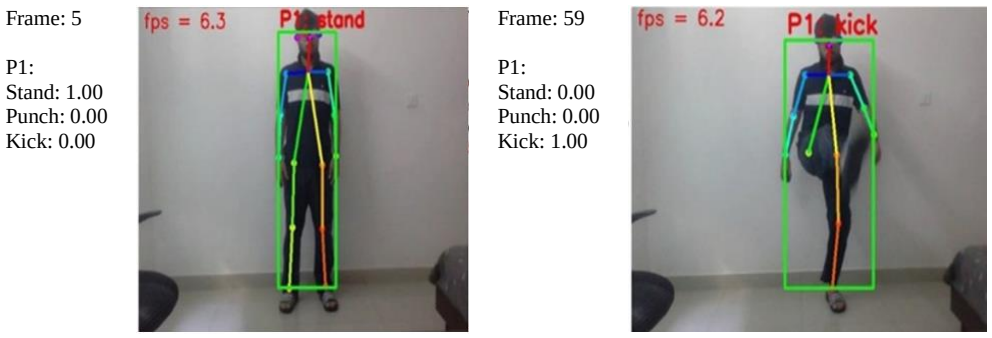
3.1. Test data

For testing purposes, video clippings were recorded. Five video clips for each class, each clip being 5 seconds long were generated. Each video clip in test dataset starts with the person in a standing position. If the video belongs to the BAD category, then the corresponding action is carried out once in the video clip. For the video belonging to the GOOD category, the subject(s) remains standing throughout the test clip. Results for the experiments have been detailed in the next two subsections. We again followed the idea of recording these test clips in diferent orientations of the person present in the video clip to account for variability in the positioning of the people in a real life video feed as shown in Figure 5.

HARecognizer converts the 5 second video clip to a folder of images on which predictions are then made. It converts those images back into a playable video. Figure 6 shows some frames from the prediction video on the test dataset. The scores on the left indicate the confidence score for each class that the model has been trained upon. The Figure 6 showcases results (a) shows where the person was standing in a front facing orientation and (b) the model correctly identified the frames where kick action was being performed.

The Figure 7 shows from a prediction video where the person has a slightly different orientation as compared to the first set. The model starts off by correctly identifying the person performing a standing action. After that the person starts to perform a punch action which is again correctly identified as punching action. However, observing the third set in the image, we see the model incorrectly identifies it as a kicking action momentarily. So, although the model identifies some sort of action occuring, it gives it a wrong label. In the next section we describe in depth on how to deal with these predictions and come up with new metrics to compare different methods for the same.

Figure 5. Different captures from the test clips showing different orientations of the actions



(a) (b)

Figure 6. Model predicting for; (a) standing position; (b) kick action

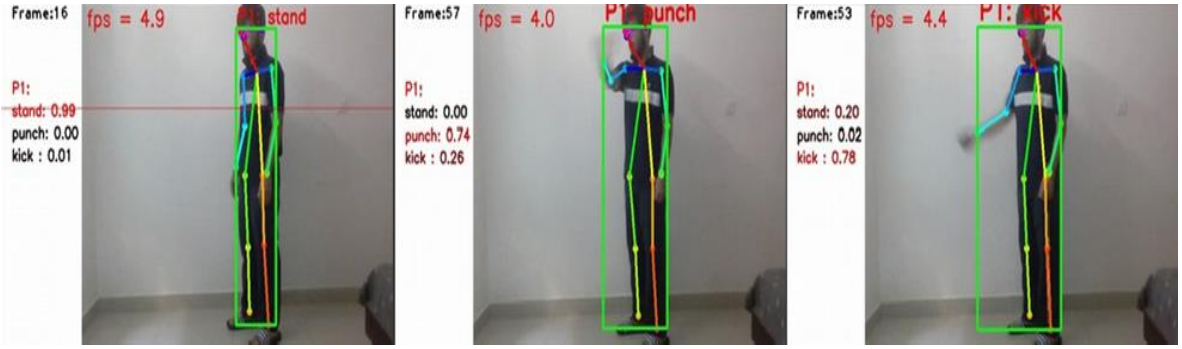


Figure 7. Model predictions on a different instance in our test dataset

3.2. Accuracy metrics

Human action recognition system converts the 5 second video clip to a folder of images on which predictions are being made. Since the sliding window size is 5 frames, no predictions are made for the first 4 frames. For each video example the system will identify the dominant action performed in that video clip. Then the number of occurrences of each predicted action class results from each frame in the following format:

$$\{\text{`'}: a, \text{`stand'}: b, \text{`punch'}: c, \text{`kick'}: d\}$$

a = number of times prediction as None, b = number of times prediction as Stand,
c = number of times prediction as Punch, d = number of times prediction as Kick

If the target belongs to the stand category, the video is marked as True Negative if $c + d = 0$ and False Positive if c+d>0.

If it is marked a true negative, the accuracy for target label stand is calculated using:

$$stand\ Accuracy = (b / a + b + c + d) * 100\ \%$$

It is then multiplied with the True Negative to get a Weighted True Negative.

$$Weighted\ TN = stand\ Accuracy * True\ Negative$$

If the target belongs to the other category (Label is punch or kick), then we mark the entire example as one count of true positive if c + d > 0 or a false negative if $c + d = 0$. However, this does not account for the actual prediction being made is correct or not depending on the target. In order to account for the actual prediction being made, we use the following formula for the accuracy of the particular example.

$$punch|kick\ Accuracy = (c/c + d) * 100\%$$

If the video is marked as true positive, we use this accuracy and multiply it with true positive to get a weighted true positive.

$$WeightedTP = punch|kick\ Accuracy\ *\ True\ Positive$$

We finally use these to define three new metrics for comparing the two different techniques on the test dataset.
Weighted accuracy:

$$[WeightedTP + WeightedTN] \div [TP + TN + FP + FN]$$

Weighted recall:

$$(WeightedTP) \div (TP + FN)$$

Weighted precision:

$$(WeightedTP) \div (TP + FP)$$

The recorded metrics mentioned above for this approach are listed in Table 2. Note that the results are rounded up to the nearest 2 decimal points.

Table 2. Table depicting weighted metrics

Metric\Technique	HARecognizer
Weighted Accuracy	89.42%
Weighted Recall	85.82%
Weighed Precision	85.82%

4. CONCLUSION

The results clearly indicate that a human pose estimation based approach gives good results when it comes to developing a system that is robust towards false positives and false negatives. It also shows that the model will perform ideally in real time environments especially when dealing with situations where the movement of humans can be erratic or unpredictable and in a environment where giving even false positives as predictions can be seen as a harmful result to the parties present inside the scene at the given moment. The weighted metrics that were used shows more reliable measure, accuracy of about 89.42% with precision of about 85.82% and shows the effectiveness of the system.

Deployment and future work: Actual deployment is another important aspect to the proposed deep learning model. Since this is a compute heavy model, it is not easy to run it on low end devices such as a raspberry-pi where the model runs independently on each device. Instead a better approach would be to stream the live video feed to an offboard server or using a cloud based system, availing the service of a cloud provider service such as Google Cloud Platform or AWS. This would transfer all of the compute needs to a separate location and allow us to install cameras with streaming capabilities to a remote server. The choice of cloud service will be based upon whichever is most cost beneficial since the models would need to be deployed on GPU-based machines to provide a relatively good performance. For some of our future work, we hope to develop a faster pose estimation model by carrying out a more extensive feature engineering process and determining what combination of features can further aid us in increasing the speed and accuracy of the system. We hope to use this and evaluate performance gains with respect to loss of accuracy in detection since the main goal here is in deploying such a system on lower end devices like raspberry pi which typically do not have a lot of compute to offer, especially for complex computations present in our deep learning models. This can ensure the real-time aspect that we are looking to achieve for our system while taking into account acceptable amounts of accuracy losses. We hope to develop more comprehensive accuracy metrics that can further serve as a benchmark when comparing different systems for real time activity recognition.

REFERENCES

[1] M. S. Iqbal, "Use of Technology Innovation Tools to combat the growing Child abuse abduction based on the case studies of Pakistan," 2018. [Online]. Available: www.academia.edu/42202082.
[2] S. Raflesia, "Geofencing based technology towards child abuse prevention," *2017 International Conference on Electrical Engineering and Computer Science (ICECOS)*, 2017, doi: 10.1109/ICECOS.2017.8167125

[3] S. Shruthi, "Detection and prevention of child abuse using IOT," *International Journal of Computer Science and Mobile Applications*, vol. 6, no. 2, pp. 139-144, 2018.
[4] A. Jatti, M. Kannan, R. M. Alisha, P. Vijayalakshmi, and S. Sinha, "Design and development of an IOT based wearable device for the safety and security of women and girl children," *2016 IEEE International Conference on Recent Trends in Electronics, Information and Communication Technology (RTEICT)*, 2016, pp. 1108-1112, doi: 10.1109/RTEICT.2016.7808003
[5] National Children's Allicance, "National Statistics on Child Abuse," 2018. [Online]. Available: www.nationalchildrensalliance.org/media- room/national-statistics-on-child-abuse.
[6] Children's Bureau, US Department of Health and Human Services, "Child Maltreatment 2018," 2018. [Online]. Available: https:// www.acf.hhs.gov/sites/default/files/cb/cm2018.pdf#page= 71.
[7] J. K. Aggarwal and Q. Cai, "Human motion analysis: A review," *Proceedings IEEE Nonrigid and Articulated Motion Workshop*, 1997, pp. 90-102, doi: 10.1109/NAMW.1997.609859.
[8] M. Ramanathan, W. Yau, and E. K. Teoh, "Human action recognition with video data: Research and evaluation challenges," *IEEE Transactions on Human-Machine Systems*, vol. 44, no. 5, pp. 650-663, 2014, doi: 10.1109/THMS.2014.2325871.
[9] Y. Li and Y. Kuai, "Action recognition based on spatio-temporal interest points," *2012 5th International Conference on BioMedical Engineering and Informatics*, 2012, pp. 181-185, doi: 10.1109/BMEI.2012.6512972.
[10] R. Poppe, "A survey on vision-based human action recognition," *Image and Vision Computing*, vol. 28, no. 6, pp. 976-990, 2010, doi: 10.1016/j.imavis.2009.11.014.
[11] X. Ji and H. Liu, "Advances in View-Invariant Human Motion Analysis: A Review," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 40, no. 1, pp. 13-24, 2010, doi: 10.1109/TSMCC.2009.2027608.
[12] D. Weinland, R. Ronfard, and E. Boyer, "A survey of vision-based methods for action representation, segmentation and recognition," *Computer Vision and Image Understanding*, vol. 115, no. 2, pp. 224-241, 2011, doi: 10.1016/j.cviu.2010.10.002.
[13] J. M. Chaquet, E. J. Carmona, and A. F.-Caballero, "A survey of video datasets for human action and activity recognition," *Computer Vision and Image Understanding*, vol. 117, no. 6, pp. 633-659, 2013, doi: 10.1016/j.cviu.2013.01.013.
[14] L. Wang, Y. Qiao, and X. Tang, "Motionlets: Mid-level 3D parts for human motion recognition," *2013 IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 2674-2681, doi: 10.1109/CVPR.2013.345.
[15] K. Alex, S. Ilya, and H. Geoffrey, "ImageNet classification with deep convolutional neural networks," Neural Information Processing Systems, vol. 25, pp 1-9, 2012, doi: 10.1145/3065386.
[16] J. Donahue *et al.*, "Long-term recurrent convolutional networks for visual recognition and description," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 4, pp. 677-691, 2017, doi: 10.1109/TPAMI.2016.2599174
[17] S. Ji, W. Xu, M. Yang, and K. Yu, "3D convolutional neural networks for human action recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 221-231, 2013, doi: 10.1109/TPAMI.2012.59
[18] E. Cippitelli, E. Gambi, S. Spinsante, and F. F.-Revuelta, "Evaluation of a skeleton-based method for human activity recognition on a large-scale RGB-D dataset," *2nd IET International Conference on Technologies for Active and Assisted Living (TechAAL 2016)*, 2016, pp. 1-6, doi: 10.1049/ic.2016.0063
[19] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3D convolutional networks," *2015 IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 4489-4497, doi: 10.1109/ICCV.2015.510
[20] I. Haroon, Z. Amir, J. Yu-Gang, G. Alexander, L. Ivan, S. Rahul, and S. Mubarak, "The THUMOS challenge on action recognition for videos "in the wild"," *Computer Vision and Image Understanding*, vol. 155, pp. 1-23, 2016, 10.1016/j.cviu.2016.10.018.
[21] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. V. Gool, "Temporal segment networks: Towards good practices for deep action recognition," *European Conference on Computer Vision ECCV 2016*, vol. 9912, 2016, pp. 20-36, doi: 10.1007/978-3-319-46484-8_2.
[22] Z. Cao, G. Hidalgo, T. Simon, S. -E. Wei, and Y. Sheikh, "OpenPose: Realtime multi-person 2D Pose estimation using part affinity fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 1, pp. 172-186, 2021, doi: 10.1109/TPAMI.2019.2929257
[23] A. Toshev and C. Szegedy, "DeepPose: Human pose estimation via deep neural networks," *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1653-1660, doi: 10.1109/CVPR.2014.214.
[24] V. Belagiannis and A. Zisserman, "Recurrent human pose estimation," *2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)*, 2017, pp. 468-475, doi: 10.1109/FG.2017.64
[25] A. Bulat, G. Tzimiropoulos, "Human pose estimation via convolutional part heatmap regression," *European Conference on Computer Vision-ECCV 2016*, vol 9911, 2016, pp. 717-732, doi: 10.1007/978-3-319-46478-7_44.
[26] X. Chu, W. Yang, W. Ouyang, C. Ma, A. L. Yuille, and X. Wang, "Multi-context attention for human pose estimation," *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 5669-5678, doi: 10.1109/CVPR.2017.601.
[27] W. Yang, S. Li, W. Ouyang, H. Li, and X. Wang, "Learning feature pyramids for human pose estimation," *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 1290-1299, doi: 10.1109/ICCV.2017.144.

BIOGRAPHIES OF AUTHORS

Prof. Srividya M. S. is an Assistant Professor at R.V College of Engineering. She has over 10 years of experience in teaching and 8 years of experience in industry. Main area of research interest is Computer Vision and Deep Learning. She has guided many UG projects and has many publications in international journals and conferences.

Dr. Anala M. R. is a Professor at R.V College of Engineering. She has over 19 years of experience in teaching. Main area of research interest is Computer Architecture, High Performance Computing, Distributed Systems and Parallel programming. She has guided 40 UG projects and 20 PG projects. She has many publications in international journals and conferences.

Mr. Chetan Tayal is a final year under graduate student at R.V College of Engineering. His professional experience lies in Machine Learning applied in Computer Vision and his main research interests are based in the fields of Deep Learning, Image Processing and Computer Vision. He loves developing and deploying applications at scale.