

CSE541 Computer Vision

Weekly Report 7

**Landing Error Scoring System for Basketball: A Computer Vision Approach**

Submitted to faculty: Mehul Raval

Date of Submission: 30-03-24

Student Details

| Roll No. | Name of the Student |
| --- | --- |
| AU2140133 | Dhairya Shah |
| AU2140159 | Aayushi Shah |
| AU2140163 | Raj Dave |
| AU2140213 | Vanaja Agarwal |

2023-2024 (Winter Semester)

### **Aim:**

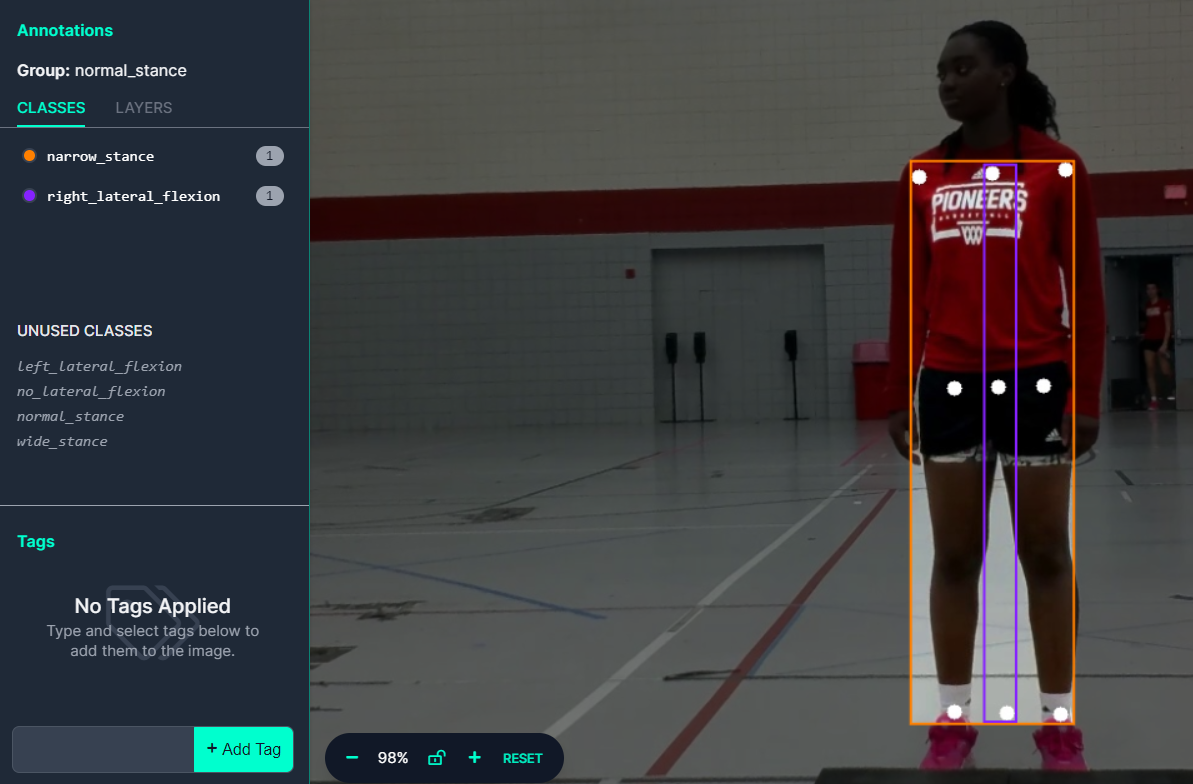
Identifying a pre-trained computer vision model or architecture specific to human activity recognition and annotating the dataset for further training and testing.

### **Introduction:**

Our primary objective is to identify a computer vision-based pre-trained model or architecture and train our dataset on it to identify and classify two specific types of errors – lateral trunk flexion and stance width – observed during an athlete's countermovement jump. Our dataset consists of manually annotated 210 frames extracted from the countermovement jump video, each labeled with the respective error classes. By implementing transfer learning on our chosen pre-trained model, we hope to speed up the learning process and make our model efficient at detecting those specific classes of errors.

### **Work Completed:**

* Annotation and labeling using roboflow



We annotated 210 frames, and we kept three classes for each of the two errors, i.e., for lateral trunk flexion, the three classes were left lateral flexion, right lateral flexion, and no flexion, and for stance width, the three classes were narrow stance width, normal stance width, and wide stance width. For two errors, we made one bounding box each. These annotated frames will be subsequently used to train our model.

* **Literature review for model selection**

CNN: By analyzing the given input image frames, pre-trained CNN will extract the low-level hierarchical features by convolutional layers, which will then be processed through the pooling layer. These initial layers of HAR models will be frozen for transfer learning.  
RNN: Compared to CNNs, RNN will also extract the temporal features. Using pre-trained LSTM-based RNNs, we could train a better model for predicting countermovement jump errors.

* **Examining characteristics and finding pre-trained different activity recognition models**CNN for Human Activity Recognition <https://github.com/aqibsaeed/Human-Activity-Recognition-using-CNN/tree/master>

This script is used to implement and train a convolutional neural network (CNN) to recognize human activities(six activity classes: walking, walking upstairs, walking downstairs, sitting, standing, or laying) based on accelerometer data.

* **HAR-stacked-residual-bidir-LSTM**

<https://github.com/guillaume-chevalier/HAR-stacked-residual-bidir-LSTMs>

This model uses a deep neural network architecture, a stacked residual bidirectional Long Short-Term Memory (LSTM) model, for human activity recognition using accelerometer data. It uses two types of datasets: one with six movement categories and another with 18 activity classes.

* **Loading MoViNett pre-trained model**

Link to the collab: <https://colab.research.google.com/github/tensorflow/docs/blob/master/site/en/tutorials/video/transfer_learning_with_movinet.ipynb>

This tutorial demonstrates how to use transfer learning with MoViNet models for video classification tasks. It involves using a pre-trained MoViNet model and creating a new model by adapting the pre-trained one.

### **Next steps and goals:**

* Transfer learning using a suitable Activity Recognition model.
* Evaluating our model using the ground truth (annotated and labeled images)

### **Conclusion:**

We have identified and annotated 210 frames from the countermovement jump video focused on detecting lateral flexion and stance width errors. We have explored a couple of pre-trained models specific to human activity recognition and are in the process of selecting a model best suited for our problem statement and then implementing transfer learning based on it.

### **References:**

1. Adrian Rosebrock, "Human Activity Recognition with OpenCV and Deep Learning," November 25, 2019. [Online]. Available: <https://pyimagesearch.com/2019/11/25/human-activity-recognition-with-opencv-and-deep-learning/> [Accessed: March 30, 2024].
2. TensorFlow, "MoViNet for streaming action recognition," [Online]. Available: <https://www.tensorflow.org/hub/tutorials/movinet> [Accessed: March 30, 2024].
3. TensorFlow, "Transfer learning for video classification with MoViNet," [Online]. Available: <https://www.tensorflow.org/tutorials/video/transfer_learning_with_movinet> [Accessed: March 30, 2024].
4. J. Brownlee, "Deep Learning Models for Human Activity Recognition," Deep Learning for Time Series, Aug. 5, 2019. [Online]. Available: <https://machinelearningmastery.com/deep-learning-models-for-human-activity-recognition/> [Accessed: March 23, 2024].