Sports Action Recognition - Final Project

Yarin Bekor, Tal Dugma, Yonatan Ashlag {yarin.bekor, tal.dugma, yonatan.a}@campus.technion.ac.il

ID: 322453176,326614617,324292143

Abstract: This report concludes our project for the "Deep Learning" course. We present a model designed to recognize sports players and their actions on the court, incorporating various training methods and pre-trained models. We achieved a 60% recognition accuracy and an inference time that qualifies as real-time recognition. Our model could be incorporated into various sports streaming platforms, including the NBA, NFL, and MLB, to facilitate a richer streaming experience and enhanced statistical analysis. We enjoyed implementing the diverse techniques we acquired throughout the semester and creating a project that combines both our academic learning and personal interests.

1 Introduction

Machine learning has made an impact on nearly every aspect of our lives, with the development of deep learning accelerating its spread. Introducing ML and DL into the sports field should begin with a foundational understanding of environmental context, which includes recognizing the positions and actions of different players, and mapping the play area's spatial boundaries.

In this project, we aim to build a model designed to detect sports players in a game and recognize their actions from raw footage. Specifically, we focus on basketball games, with the intention of recognizing the most common actions including shooting, dribbling, and more.

Achieving such a model can benefit in various domains, including live streaming, statistical analysis, player acquisitions, game-plans strategies, and training plans, and can even be utilized to design sports betting platforms.

Human detection has evolved rapidly with the introduction of deep learning in the 2010s, particularly with Convolutional Neural Networks (CNNs) [1], which have since become the backbone of modern human detection systems [2]. This period marked the beginning of human detection technologies being robust enough for practical, real-world applications, leading to their integration into various industries and services.

Action recognition has progressed from simple motion detection [3] to understanding complex activities [4], thanks to deep learning. Early methods relied on manual feature identification, but the advent of CNNs and Recurrent Neural Networks (RNNs) in the 2010s revolutionized the field [5]. These models allow us to recognize a broad spectrum of human actions automatically, and with the improvement of these, such as Long Short-Term Memory (LSTM) networks [6], action recognition now supports diverse applications.

2 Method

Our model is based on 2 main stages of processing. The first, player detection (Figure 1), refers to the process of detecting players on the court using computer vision techniques, extracting physical features of their movements, and tracking them through the video. The second stage of our model, action segmentation (Figure 4), refers to the process of recognizing the actions of each player individually, given the data extracted from the player detection stage. Combining those stages will yield the full pipeline of our model, which gets a raw sports video as input and returns the action segmentation for each player.

2.1 Player detection

In The first stage of training, we are using a pre-trained object detection model, YOLOv8, [7] which can be flagged to detect specific object classes (in our case, humans). The output of this pre-trained model is a bounding box containing only one person, for each person recognized in the raw footage. We are reshaping all the bounding boxes that were detected by padding them to match the size of the largest bounding box, due to the requirements in the following stages.

2.2 Features and skeleton extraction

The first stage of training also involves the extraction of data from the detected human bounding boxes. We here focus on two techniques, Feature extraction, and Skeleton extraction. Feature extraction refers to the use of a pre-trained model, using layers of convolutions, pooling, and other techniques that perform extraction of features from the image. To perform this conversion, we tried utilizing the Mobile-net3 pre-trained model. We replaced the last two layers of the pre-trained model with a two-layer MLP, which outputs a vector of the dimension of the number of actions of our recognition task, resulting in a model that gets a video as input and returns a representing vector. The second technique,

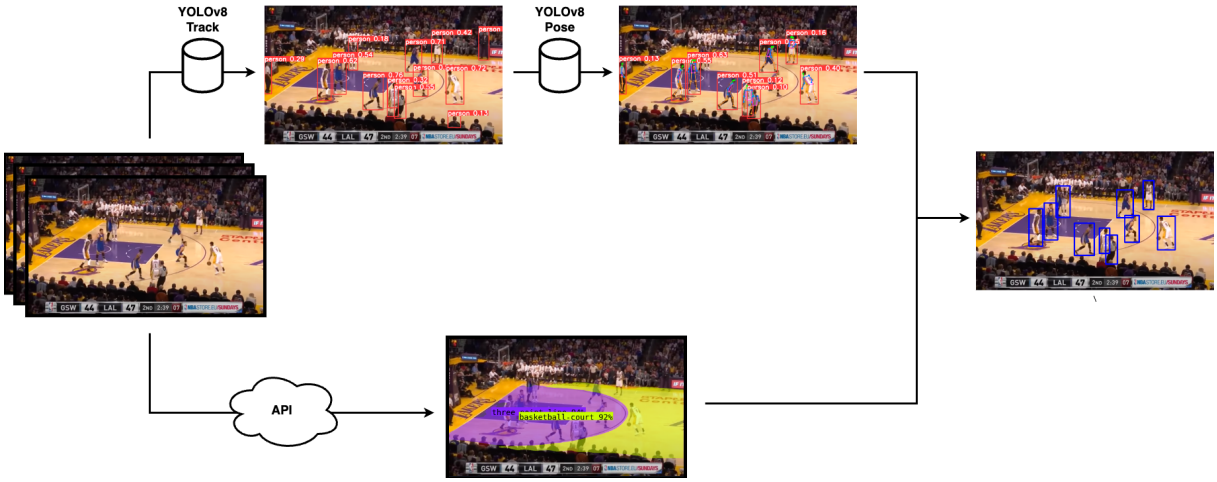

Figure 1: *The first stage of training, starting from an input video, we are extracting the bounding boxes and the skeletons of the different players using two flavors of the YOLOv8 pre-trained model: the Track model for bounding boxes, and the Pose model for extracting the skeletons. Then, using another pre-trained model, we segment the court and keep only the players who are inside.*

skeleton extraction, detects the different joints of the human in the frame, which are the 17 key points of his body, in the format of nodes and edges. To perform this prediction we are using again the pre-trained model, YOLOv8-pose. We then converted the predicted nodes into (x,y) coordinates, representing the nodes' location on the padded bounding box grid.

2.3 Action segmentation

The second part of the pipeline involves training an LSTM with an Attention mechanism, to segment the different actions during the video successfully. The LSTM receives a tensor representing the locations of the key points of a player over the frames of the video, and predicts the action performed by the player. We trained a model containing 2 stacked layers of LSTM with an attention mechanism, with a hidden layer of size 126, adding to a total of 224,658 parameters.

Our motivation was to perform an ensemble of models- the skeletal-based model and the image-based model, which can leverage the strengths of each model. An image-based model has the potential to be accurate in cases where a ball in the frame differs from two labels, while the other model will struggle. In our case, for this specific dataset, the image-based model, and others similar to it, didn't perform well and therefore was not included in the ensemble.

To solve the imbalanced data problem, we used Weighted Cross Entropy loss, which is weighted inverse to the frequency of every label. Using this loss criterion, the model can keep learning without over-labeling predictions as the most frequent labels (and by that overfit to the training data). As we have seen in the lecture, when using this loss we sometimes can see the loss on the validation set rising, and should still train because the accuracy and F1 will also rise (Figure 2).

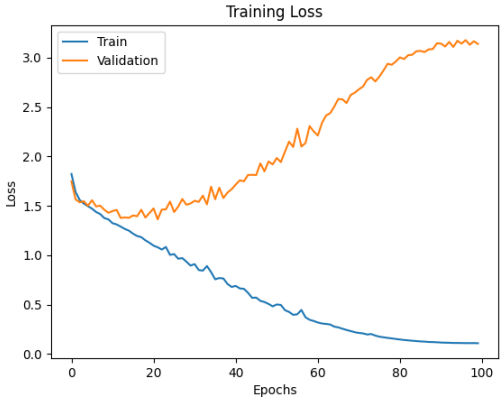

Figure 2: *Loss over Epochs*

To improve our learning we also used a cosine learning rate scheduler (Figure 3).

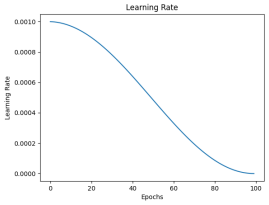

Figure 3: *Learning Rate over Epochs*

2.4 Binary Court Segmentation

To make sure our model only detects the players' actions and ignores the other humans who appear in the video.

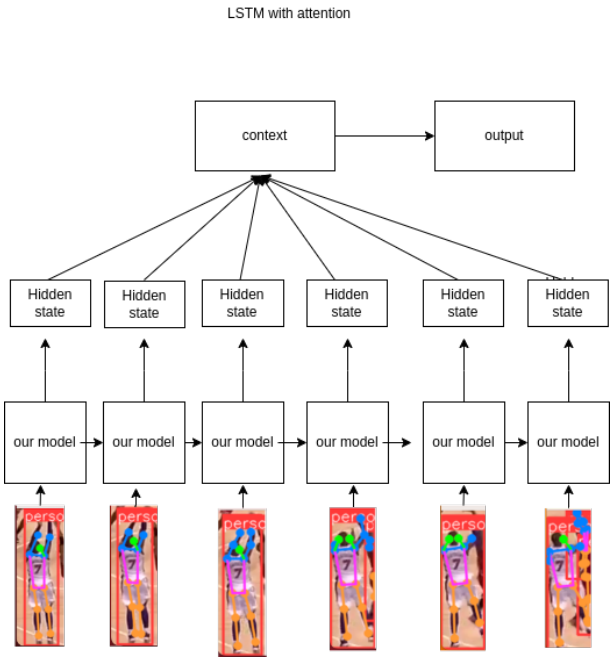

Figure 4: *LSTM with attention model*

To solve this problem, we classified each pixel as an in-court or not. To perform such segmentation, we used a pre-trained model, found in RoboFlow, a web page that hosts pre-trained models [8]. After extracting the court mask from each frame, we ensure that only specific bounding boxes will be labeled as actual players.

Algorithm 1 Check if a player is on the court. court_mask is a matrix of 1's and 0's, based on the API's prediction of whether the pixel i,j is in the court.

$Bbox\_lowest\_row \leftarrow (box[0][:], box[1][-1])$
for (x, y) in $bounding\_box\_lowest\_row$ **do**
  if $court\_mask(x, y) == 1$ **then**
    return True
  end if
end for
return False

Since our goal is to detect actions in the game, we only care about the players in the frame. We solve this by using a model that outputs a binary court mask. During inference, we first get the bounding boxes for all characters in the frame by our player recognition model and then apply our action segmentation model only on the bounding boxes that overlap with our court mask. By that, we reduced computation runtime and better visualization for the final model.

3 Dataset

To train this model, we supervised it using the "space-jam" dataset, created by Dr. Simone Francia during his PhD research [9]. The dataset consists of 37,085 short videos, each displaying a single player performing a specific action. Each video is labeled by one of the 10 available actions, based on the action occurring in it. However, there is a significant imbalance in their distribution, resulting in a biased learning process. Therefore, we randomly removed 30% of the "walk", "no_action", and "pick" actions, and added 50% of augmented data for each of the less expressed actions, resulting in better-distributed data (Figure 5).

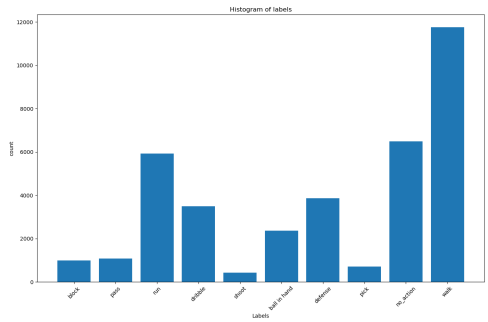

Figure 5: *Distribution of the different actions in the raw dataset (after augmentations)*

To simplify the labels of the dataset and improve training for the model, we decided to combine the similar actions "walk", "no_action" and "pick" as one label - "no_action" (Figure 6).

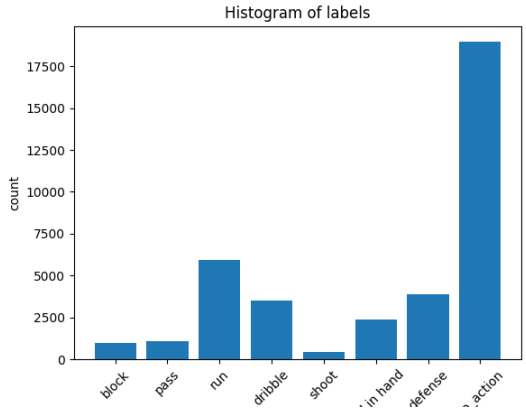

Figure 6: *Distribution of the different actions in the dataset after the removal of over-expressed actions and combining similar labels.*

4 Results

Our model reaches 60% accuracy, 60% F1-score, and struggles to detect several labels (Figure 7), especially when actions look similar if a ball is not considered (like shooting and blocking, running and dribbling, etc.).

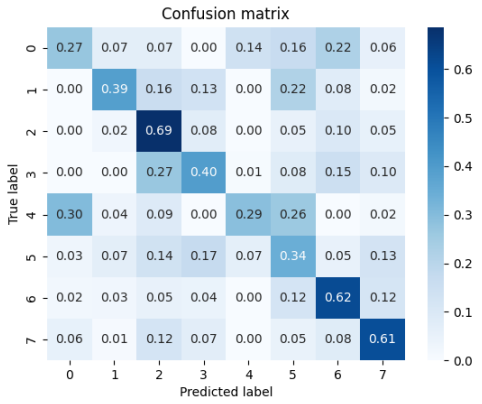

Figure 7: *Confusion Matrix using labels: {0: "block", 1: "pass", 2: "run", 3: "dribble",4: "shoot",5: "ball in hand", 6: "defense", 7: "no action"}*

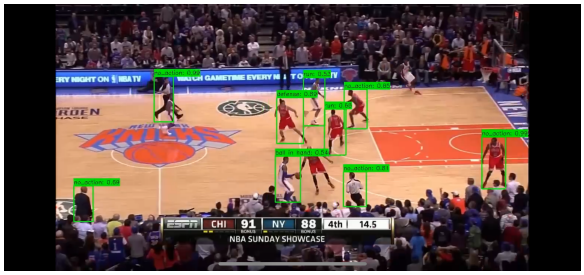

Figure 8: *Examples for the output of the full model*

5 Ethical statement

Stakeholders:

Professional sports teams, professional sports players, sports trainers, and betting agencies.

Implications:

Professional sports teams could use the model to get advanced statistics for opponent players or for players they are interested in signing. Sports trainers could use this to analyze the performance of their trainees and track their improvement. Betting agencies could use the model to introduce new bets based on the automated stats.

Ethical Considerations:

Potential stakeholders that choose to rely on such models to derive conclusions should take into account any biases the model might have collected during the learning. In cases where the predictions' accuracy is critical, such as in betting platforms, relying on the model could be dangerous for the stakeholders. In addition to that, for models that predict the actions of humans, it is important to make sure that the model does not base the inference on parameters that shouldn't be taken (such as skin color, race, etc).

6 Discussion

This project showcases the integration of deep learning with sports analytics, presenting a novel model for real-time sports action recognition. By leveraging advanced object detection algorithms and convolutional neural networks, such as YOLOv8, we developed a system that can identify and analyze players' actions. This advancement not only enhances the analytical capabilities within sports but also enriches the viewer's experience by providing detailed insights into players' performances in real-time. The application of pre-trained models for player detection and action recognition reflects a strategic approach to efficiently solve complex problems, highlighting the project's contribution to both theoretical and practical aspects of sports analytics and machine learning.

Looking ahead, there is ample scope for further refining the model to encompass a wider range of sports and actions, thereby broadening its applicability and impact. Future work could explore the integration of temporal dynamics to better understand the sequences of actions, offering deeper insights into players' strategies and performance patterns. Additionally, expanding the dataset to include more diverse sporting contexts and actions could enhance the model's accuracy and robustness, making it a more powerful tool for coaches, players, and fans alike. This project lays a solid foundation for

future explorations in the convergence of deep learning and sports, promising exciting advancements in sports technology and analytics.

7 References

References

[1] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. "ImageNet Classification with Deep Convolutional Neural Networks". In: *Advances in Neural Information Processing Systems*. Ed. by F. Pereira et al. Vol. 25. Curran Associates, Inc., 2012. URL: https://proceedings.neurips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.

[2] Md. Milon Islam et al. "Human activity recognition using tools of convolutional neural networks: A state of the art review, data sets, challenges, and future prospects". In: *Computers in Biology and Medicine* 149 (Oct. 2022), p. 106060. ISSN: 0010-4825. DOI: 10.1016/j.compbiomed.2022.106060. URL: http://dx.doi.org/10.1016/j.compbiomed.2022.106060.

[3] Zihan Wang et al. *Deep Neural Networks in Video Human Action Recognition: A Review*. 2023. arXiv: 2305.15692 [cs.CV].

[4] "Dynamic Semantic-Based Spatial Graph Convolution Network for Skeleton-Based Human Action Recognition". In: 38 (Mar. 2024), pp. 6225–6233. DOI: 10.1609/aaai.v38i6.28440. URL: https://ojs.aaai.org/index.php/AAAI/article/view/28440.

[5] Sepp Hochreiter and Jürgen Schmidhuber. "Long Short-term Memory". In: *Neural computation* 9 (Dec. 1997), pp. 1735–80. DOI: 10.1162/neco.1997.9.8.1735.

[6] "Co-Occurrence Feature Learning for Skeleton Based Action Recognition Using Regularized Deep LSTM Networks". In: 30 (Mar. 2016). DOI: 10.1609/aaai.v30i1.10451. URL: https://ojs.aaai.org/index.php/AAAI/article/view/10451.

[7] Ultalyrics. *Yolov8*. https://github.com/ultralytics/ultralytics. Open Source Models. Nov. 2023. URL: https://github.com/ultralytics/ultralytics.

[8] Jonatan Beiruty. *court detection Dataset*. https://universe.roboflow.com/jonatan-beiruty/court-detection. Open Source Dataset. visited on 2024-03-31. Dec. 2023. URL: https://universe.roboflow.com/jonatan-beiruty/court-detection.

[9] Simone Francia. "Classificazione di Azioni Cestistiche mediante Tecniche di Deep Learning". PhD thesis. Apr. 2018.