October University for Modern

Sciences and Art

Faculty of Computer Science

Graduation Project

Basketball Action Recognition

Supervisor:

DR. Wael Farouk Mohamed El-Sersy

Name: Mariam Adel Muhammed Mazhar

ID: 220737

Chapter 1: Introduction

1.1 Introduction

The fast expansion of basketball leagues such as the NBA, NCAA, and CBA has resulted in a considerable increase of basketball video material, enhancing the possibility of applying computer vision technology for sports analytics. The automated detection of player movements in these recordings can help analysts assess player and team performance and make more informed game decisions. Although single-player action recognition has been widely studied, two-player or more interaction analysis remains open research subject due to the spatiotemporal complexity of team sports. The Space Jam dataset, which contains tiny video clips of individual players cropped from whole-screen footage, is used in this work to investigate basketball action identification.

The field consists of player, player activity, and ball detection using advanced deep neural network learning patterns such as 3D Convolutional Neural Networks (3D CNN), Long Short-Term Memory networks (LSTM), Slowfast, and i3d (Inflated 3D Networks). These are intended to efficiently handle both spatial and temporal features, allowing for accurate action recognition. In addition, a post-processing method is proposed to assess recognized activities in respect to player positions and interactions. All highly competitive basketball clips are now produced with a single camera, making it difficult to examine simultaneous moves.

Currently, most competitive basketball videos are shot with a single camera, making it difficult to examine numerous simultaneous movements. Recent breakthroughs in deep learning, such as the YOLOv11 model (Figure 1), have improved player detection and tracking in single-feed videos. However, existing methods often struggle with recognizing complex interactions between players or people on the background and handling occlusions or overlapping actions.

Given recent advances in deep learning, such as the YOLOv11 model shown in Figure 1, single-feed video identification and player tracking have improved. However, existing approaches struggle to identify complicated interactions between players or persons in the background, as well as deal with occlusions or overlapping motions.

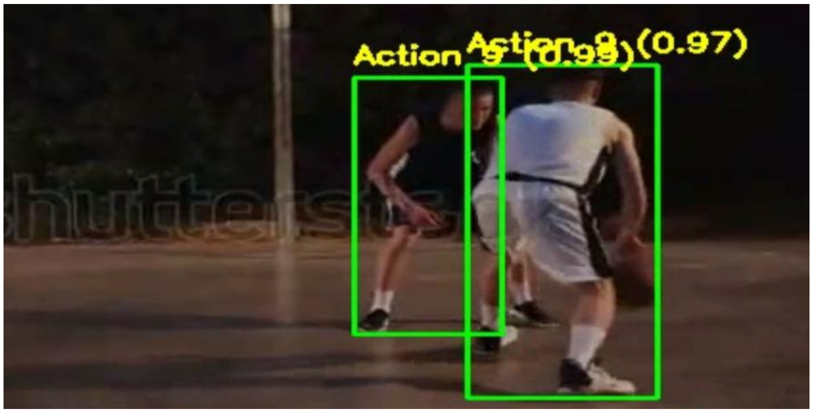

Figure 1: Player detection and action classification using YOLOv11 and 3D CNN models

While previous approaches show promise for identifying players and recognizing single actions, they fall short when it comes to analyzing multi-player interactions and considering contextual elements such player placements and team movement. This study aims to address these constraints by using spatiotemporal models and post-processing to improve action identification accuracy in basketball footage.

1.2 Problem statement

Current automatic basketball action detection systems have a number of technological limitations when employed with actual game footage. This covers issues in:

- simulating rapid, non-linear motion patterns generated by fast-paced video games.
- separating occlusions and overlapping players, especially with single-camera streams.
- keeping the temporal dependence of complex processes requiring several frame sequences.
- correctly recognizing multi-layered, intricate events (such as a pass vs a fake pass).

In summary, traditional computer vision algorithms and basic frame-by-frame classification models give poor spatiotemporal rseasoning, which explains why they are unsuitable for the present purposes: an all-encompassing deep-learning system that can

- Player tracking and recognition stay robust even in occlusion or poor visibility.
- Accurate spatiotemporal action classification with appropriate models such as 3D CNNs, LSTM, I3D, and SlowFast networks.

- Ball identification and analysis of player interactions can be considered some examples of context-aware post-processing.
- This would potentially improve the accurateness of such basketball video analytics while the system would render the whole annotation process automated, which is beneficial for performance analysis, strategic coaching, and in-the-moment decisions.

1.3 Objectives

The major objectives of this project are to develop a reliable system that:

1. Localizes basketball players from short video clips.
2. Accurately classifies their actions, including dribbling, passing, shooting, and defending.

3. Applies contemporary machine and deep learning models such as CNN, LSTM, and 3D-CNN+LSTM architectures to efficiently handle spatial and temporal information.
4. Detects basketballs in video frames to improve action recognition.
5. Rates the player's performance as either "Qualified," "Not Qualified," or "Fair" based on the player's actions and effectiveness.

1.4 Motivation

– Motivation for others

Proposed research is highly important to automate sporting analytics and officiating with respect to recognition of player actions and interactions within video footage pertaining to basketball. This research aims to fill a significant gap in current systems regarding multi-player dynamics and context analysis, thereby contributing an additional layer of understanding to team performance. There also lie possibilities for fundamental changes to how coaches, analysts, and referees use video data in the decision-making and strategizing processes in competitive sports.

—Motivation for the researcher

As a researcher, I am truly passionate about finding solutions for real-life problems with the help of modern technology. I am extremely excited about this project as it brings together my interest in computer vision and deep learning with the fast-paced environment of sports. The very ability to solve the complexities of multi-player interaction and spatiotemporal analysis would thereby hopefully push towards further developments in future AI applications. And this is a great opening to change fundamentally how the game is perceived and lived.

1.5 Thesis layout

As follows:

- The introduction chapter, which provides an overview of the background, statement of the problem, objectives, motivation, and layout of the thesis.
- This is followed by background literature review and-related studies in action recognition, along with overviews of used models and datasets.
- The materials and methods chapter provide details on the materials used in this project, such as the dataset, tools, and development environments.

Chapter 2: Background and Literature Review

2.1 Background

Recognizing actions from videos is performed by recognizing and classifying the behaviors or actions from sequences of frames. Therefore, it is necessary to extract the spatial features (what is happening in a frame) and the temporal features (how things change over time). Here, various deep learning methods are used to work with those challenges by putting together the computer vision techniques with temporal modeling for effectively recognizing basketball-related actions.

2.1.1 3D CNN:

Another fundamental method is the use of 3D convolutional neural networks (3D CNNs). 3D CNNs, in contrast to the more classical 2D CNNs that process frame by frame, are indeed used to analyze sequences of frames whereby convolutions across spatial dimensions as well as the temporal axis are performed. This enables the model to understand motion and actions in the timeline of short video clips. A custom-designed data generator has been used to initialize the feeding of video clips to the model reasonably during the training.

2.1.2 CNN + LSTM:

The CNN+LSTM architecture in addition to the 3D CNN approach was applied. This architecture applies a CNN for frame-level feature extraction, which is then put into the LSTM so that temporal dependencies can be captured. The LSTM unit thus helps the model build an understanding of how player actions change over several frames, thus increasing performance in cases where actions cannot be determined from a single frame.

2.1.3 Slow-fast:

Implemented in the project is the SlowFast network, another advanced method that employs a two-stream architecture to perform processing of visual information at different temporal resolutions. The slow pathway efficiently infers semantics at low frame rates, while fast pathways capture motion dynamics at a higher frame rate. Such hybrid architecture efficiently stitches together rapid-motion handling, particularly during basketball instances.

2.1.4 YOLOv11:
The YOLOv11 object detection model was adopted for player detection, allowing real-time player identification in video frames. The action recognition module integrated with player localization through YOLOv11 output using the 3D CNN model. This integration allows players to be detected while simultaneously classifying their actions using cropped video clips focused on the detected player regions.

2.1.5 Roboflow:
Roboflow was used to annotate a dataset for detecting other basketball-specific features such as the ball and the hoop. These annotations were used to train YOLOv11 to detect and track the ball and hoop as the match proceeded. The system could be used for additional features, such as detecting goals and tracking the interactions of players with the ball or the hoop.

2.2 Previous work:

The study critically analyzes and evaluates the relevant literature underpinning current developments. An individual work is discussed in terms of its conceptual approaches, technical methodologies, dataset characteristics, and evaluation results.

2.2.1 Research 1: Basketball Technique Action Recognition Using 3D Convolutional Neural Networks

Authors**:** Jingfei Wang, Liang Zuo & Carlos Cordente Martínez

2.2.1.1 Strategy & Structure

This paper deal with Basketball activities that are being classified using 3D Convolutional Neural Networks (3D CNNs) for dynamic object behaviors. The procedure is learning complex spatiotemporal characteristics using stacks of video frames, transforming actions from simple object identification to dynamic behaviors from an object. The structure incorporates standard 3D CNNs while leaving video segments in temporal convolutional filters. Multiple convolutional and pooling layers capture motion and structural characteristics. Custom preprocessing is designed specifically for basketball video content, ensuring strong domain adaptation capabilities.

2.2.1.2 Data

The dataset used for this study comprises cropped video parts of basketball players labeled in reference to different techniques. The size of the dataset is not provided explicitly, but it comprises short video parts of basketball techniques recorded from actual instances in the basketball field. The controlled recording took place with minimal occlusions to capture the player's intention to perform sport movements.

2.2.1.3 Method Evaluation

The strength of this work is that the use of 3D convolutional layers is by default in keeping with such basketball actions due to which it can be expected to improve distinguishing between similar-looking frames by motion information. However, a significant limitation in generalization is that the model has only been tested on single action by a single player from an isolated clip, hence it shall not be a valid argument when concerning its application and scalability in a multiplayer context or real-time applications encompassing multiple players.

2.2.1.4 Results Evaluation

The findings confirm that the recognition of basketball actions is of high appeal with 3D CNNs. The model performed extremely well in its classification task in the customized dataset, hence establishing the feasibility of affording spatiotemporal deep learning models in sports action recognition.

2.2.2 Research 2: Real-Time Basketball Shooting Action Recognition

Author: Tianyu Fan, KTH, School of Electrical Engineering and Computer Science (EECS)

2.2.2.1 Strategy & Structure

The main concern of the research is the real-time recognition of shooting actions using a lightweight deep learning pipeline. It is proposed that striking a balance between accuracy and computational efficiency renders the system deployable into a live game environment. The strategy, therefore, involves an explicit pose change and key motion phase definition of shooting actions instead of general classification.

In distinguishing shooting actions from non-shooting movements, the model integrates pose estimation as an input and uses CNNs as the output classifier, combining spatial detection and temporal phase tracking by utilizing open-source toolkits like OpenPose for skeleton tracking. The approach is novel in the sense that it presents very low latency in a real-time environment, thus making the triggering of action detection feasible.

2.2.2.2 Data

The dataset consists of video clips of basketball players performing different actions, with annotations marking the beginning and end of shooting trajectories. The data are of moderate size but are annotated with fine-grained motion phases allowing the model to focus on subtle transitions of the pose. Videos were obtained from single-angle recordings typical of training footage or broadcast games.

2.2.2.3 Method Evaluation

A great strength of the method being proposed is its focus on real-time use through efficient architectures and filtering based on poses to reduce processing overhead. The method expresses some limitation in being heavily dependent on accurate pose estimation, since occlusions or poor lighting may indeed detrimentally affect skeleton tracking, leading to misclassifications or missed detections.

2.2.2.4 Results Evaluation

The system performs real-time with fair accuracy in recognizing shooting actions. The inquisitive pose tracker when coupled with the proposed classifier renders a practical solution for live feedback in training or game scenarios.

Chapter 3: Materials and Methods

3.1 Materials and Methods

Currently present in this part of the project are materials, data, tools, and environment, as well as methodology development in the creation of the entire basketball action recognition and tracking system.

3.1 Materials

3.1.1 Data

There were two main datasets used for different purposes in this project:

Dataset 1: SpaceJam Dataset

- Source: Downloaded from GitHub - SpaceJam Dataset
- Format: MP4 video clips and accompanying .npy joint files, with action annotations stored in a JSON format.
- Structure: The original dataset consists of:

1. Video and joint data
2. annotation_dict.json (action labels per clip)
3. labels.json with 11 class mappings: "0": "block", "1": "pass", "2": "run", "3": "dribble", "4": "shoot", "5": "ball in hand", "6": "defense", "7": "pick", "8": "no_action", "9": "walk", "10": "discard" .

- Size: Ultimately 4,424 labeled video clips were used after preprocessing and disbanding joint files.
- Preprocessing: Joint files removed not meant for modeling were split into test and training videos.
- What is it about then? Purpose was for action classification with deep learning models.

Data Set 2: Roboflow Dataset (Custom Ball)

Generated: Manually by Roboflow itself using the software.

Format: Extracting frames out of a 13-second-long basketball video played at 30 frames per second, resulting in a collection of around 390 annotated frames.

Annotation: Bounding boxes drawn around ball and hoop.

It serves the purpose of using object detection and tracking and particularly training various models such as YOLOv11 for real-time detection.

3.1.2 Tools and Libraries

The system was built with Python and an amalgamation of an open-source library for deep learning, data preprocessing, and visualization:

- Programming Environment:
  Python: The language used for implementing preprocessing, model development, and evaluation pipelines as a core language.

- Deep Learning Framework:
  TensorFlow 2.x/Keras: To build, train and evaluate deep learning models. A high-level API by Keras is for building 3D CNN and hybrid models (e.g., SlowFast, CNN-LSTM).

- Video and Image Processing:
  OpenCV (cv2): Involved for reading videos, extracting frames, resizing them, and providing preprocessing steps before model input.
  Matplotlib: To visualize the model performance during training.

- Data Handling and Utilities:
  NumPy: For the manipulation of multi-dimensional arrays and matrix operations.
  JSON: To read and manipulate annotation and label files.
  tqdm: Added for progress bars to keep an eye on long preprocessing or training loops.

3.1.3 Environment

- Hardware:
  Google Colab Pro with NVIDIA A100 GPU: For faster model training and testing with real-time inference.
- Deployment:
  Cloud-based: Colab Pro for easy GPU acceleration and library dependencies.
  Reason: For expedited training, real-time inference, and scalability for large video workloads.

3.2 Methods

3.2.1 Overview of the System Architecture:

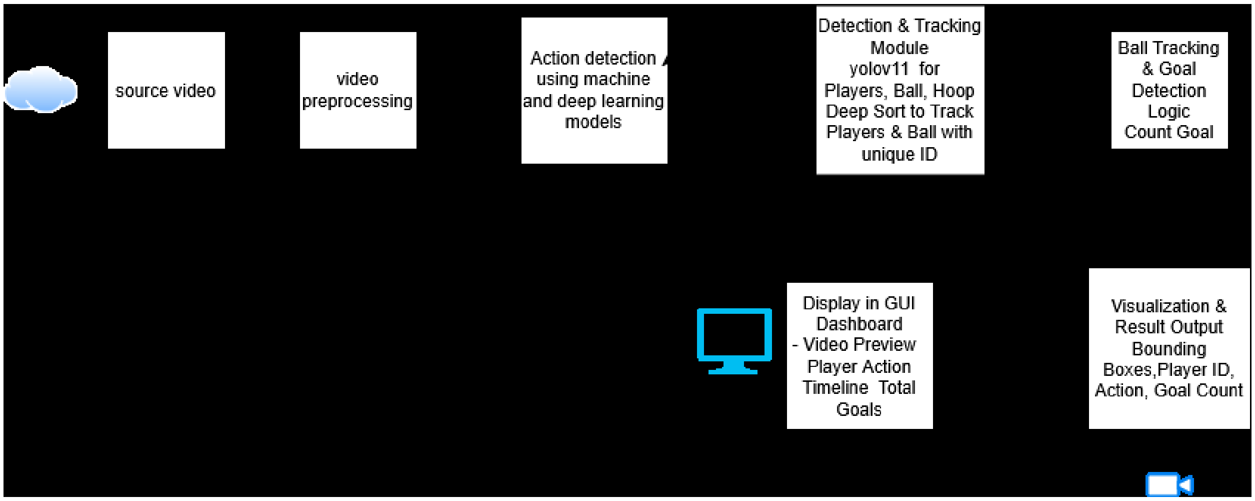
source video
video preprocessing
Action detection using machine and deep learning models
Detection & Tracking Module yolov11 for Players, Ball, Hoop Deep Sort to Track Players & Ball with unique ID
Ball Tracking & Goal Detection Logic Count Goal
Visualization & Result Output Bounding Boxes,Player ID, Action, Goal Count
Display in GUI Dashboard - Video Preview Player Action Timeline Total Goals

Figure 3.1, system architecture

As shown in Figure 3.1, our system has a modular pipeline architecture where every module is responsible for specific stages of pipeline processing-from input to visualization end. This pipeline is scalable, modular, and suitable for real-time basketball action analyses.

The whole methodology can boil down into the following steps:

1. Source Video Input

First video source into the system is raw basketball match video. These videos are typically collected or downloaded elsewhere for training or inference.

2. Video Preprocessing

- The videos are broken down into individual frames using OpenCV.
- The frames are then resized to standard sizes for deep-learning model input.
- For action classification (SpaceJam dataset), each video is sliced into short clips (usually 0.5 - 1 seconds).
- Data that is not needed (like joint files) is cleared to boost processing speed.

3. Action Detection by Deep Learning

- The SpaceJam dataset is for training the convolutional neural network (CNN) model.
- The classifier predicts one out of 11 predefined basketball actions:

{block, pass, run, dribble, shoot, ball in hand, defense, pick, no_action, walk, discard}

- Each short video is labeled with one action class for supervised learning.

4. Detection and tracking system:

Multi-object image detection and tracking systems:

- Custom train-train YOLOv11 for detection of:
  -Basketball player.
  -Basketball.
  -The basketball hoop.
- In the system, deep sort algorithms are used, which give IDs to each object invariance for tracking purposes from each of the frames.

5. Ball Tracking and Goal Detection.

Goal detection is done with a specific logic module analyzing object trajectories:

- The movement of the ball towards the hoop is being tracked by the system.
- With respect to the hoop, the alignment and position of the ball confirm a goal.
- Counter counts the number of goals detected in the video.

6. GUI Dashboard Display

An interactive Graphical User Interface (GUI) has been developed for viewing results in real-time. The dashboard offers an interface for users to:

- Upload a basketball video.
- Click a "Detect" button to start processing.
- Preview the annotated output of the video overlaying player and ball bounding boxes, action labels and unique player IDs, goal detection, and the current goal count.

7. Final Output

The final video outputs:

- Accurate bounding boxes for players and the ball, correctly labeled.
- Action classification results.
- Identification for the ball.
- Counting goals during the match's lifetime.