Volleyball Game Analysis Using Computer Vision Algorithms

Marko Plankelj
marko.plankelj@student.um.si
University of Maribor
Faculty of Electrical
Engineering and Computer Science
Maribor, Slovenia

Uroš Mlakar
uros.mlakar@um.si
University of Maribor
Faculty of Electrical
Engineering and Computer Science
Maribor, Slovenia

ABSTRACT

In recent years, modern technologies have made sports more accessible to a wider audience by providing interactive data during broadcasts, reducing the risk of human error, and enhancing athletes' performance through real-time analysis and targeted training insights. This paper combines theoretical and practical approaches by developing an application based on specific convolutional neural networks for volleyball court detection and ball tracking. The results demonstrate the potential of advanced video analytics in sports, allowing users to explore the opportunities of modern technology in improving sports performance.

KEYWORDS

computer vision, convolutional neural networks, perspective transformation, volleyball, web application

1 INTRODUCTION

For people, sport has always been a form of relaxation, socializing, an opportunity to find new friendships with proven beneficial for health. The World Health Organization recommends playing sports as part of a healthy lifestyle in a program that aims to bring people closer to physical activity as an opportunity for a healthier, happier and more productive life [11]. On the other hand, we monitor the development of modern technologies and their use in a wide variety of fields, including sports, from day to day.

Sport and modern technologies such as computer vision and machine learning were completely opposite terms a few years ago, but nowadays we can hardly imagine watching or participating in sports without the inclusion of modern technologies[5]. Despite many challenges, such as poorer video quality or overlapping players, the use of systems for detecting the field, players and their actions, and tracking the ball during play is becoming more common. Whether it's a real-time analysis of the opposing team, which helps coaches find winning tactics, or simply watching a sports broadcast, during which the directors serve interactive slow-motion footage of the actions and statistical data about the players. By using them, they want to prevent controversial situations in matches, improve training and competitor analysis, predict loads in training and matches with the aim of preventing injuries, and improve the experience of spectators with analysis before, during and after the match.

2 PROBLEM DESCRIPTION

The main objective of this paper was to develop a web application that enables the users to select a video of a volleyball game and then, with the help of computer vision algorithms, automatically detect the volleyball court and analyze the trajectory of the ball, and displaying the results through a perspective projection of the court. The utilized computer vision algorithms in this paper were convolutional neural networks (CNN).

The addressed challenge can be broken down into four steps: 1) preparation of the training data, which involved collecting, preprocessing and labeling the data, which we later augmented with the aim of increasing the diversity and volume of the dataset, 2) implementation and learning of a CNN based on the prepared dataset, 3) Perspective transformation of the volleyball court, 4) Development of the online applications and integration of the trained CNN models in connection with a perspective projection to display the final results.

3 RELATED WORK

The use of modern technologies is no longer limited to pilot projects and events of lower ranks, but is gaining ground at the highest sporting events in the world. At the Olympic Games in Paris, in collaboration with Intel, technologies were introduced to improve the experience of participants and spectators. The International Olympic Committee presented a program to use artificial intelligence in sports to improve athlete performance and spectator experience [6].

At the 2022 FIFA World Cup in Qatar, semi-automatic technology was used to check prohibited positions. The technology uses twelve cameras placed under the top of the stadium to calculate the position of twenty-nine key points on each player fifty times per second. To accurately detect the impact of the ball, they use an IMU sensor, which is placed in the middle of the ball and sends data five hundred times per second [4].

In sports such as tennis and volleyball, the Hawk-eye system has been used for many years to track the path of the ball and determine its position with the help of high-speed cameras placed around the playing surface. It identifies the pixels in each frame that correspond to the ball and then compares its position using at least two image frames recorded from different camera angles to confirm or correct the position accordingly [10]. As already mentioned, even in volleyball, as in other sports, the introduction of advanced technologies is not an exception. The Balltime Platform with Artificial Intelligence Volleyball AI (VOLL-E) divides the volleyball game into segments, facilitating match analysis and player preparation. Using a CNN model that has been trained on numerous volleyball videos, it enables automatic detection of the ball and players on the court and labeling with bounding boxes. Based on the recognized positions of the ball and players, the platform recognizes game actions such as reception or defense and automatically determines the direction of the attack and displays it visually. It also calculates the ball speed and lets the users sort game elements by individual and integrate with YouTube to share clips [2].

Similar functionality to that provided by the Balltime platform described above is also provided by an Apple mobile application called Avais, which, with monitors and analyzes the volleyball game in real time. As a result, the users avoid waiting while loading a video of a volleyball match and can obtain data for analysis at the same time [1].

4 IMPLEMENTATION

The final solution was implemented in several steps using various technologies. To ensure accessibility across different devices and locations, a web application was developed, leveraging trained neural network models for detecting volleyball courts and tracking volleyball movements through image frames.

4.1 Collection, preprocessing, labeling, and augmentation of training data

We started with collecting, preprocessing, labeling, and augmenting the obtained training data. Primary data collection was performed using freely available data on the internet. Due to a lack of sufficient data, we added our own recordings from volleyball matches to the training set. We implemented a script in Python to convert video sequences into image frames and saved them in JPG (Joint Photographic Experts Group) format, one per second. Due to the different sources of training data and their dimensions, we unified their dimensions.

Once the dimensions of the image frames were standardized, we proceeded with the labeling of the training data, using two separate methods. For the first method, we manually selected six points on each image frame, and then, for each of the six selected points, recorded the x and y coordinates in JSON (JavaScript Object Notation) format for later use, as shown on Figure 1.

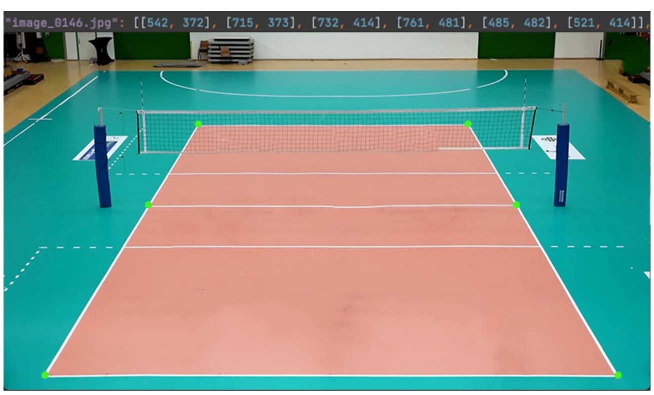

Figure 1: Manual labeling of training data points.

As for the second technique for labeling the data used in the training set for ball detection, we utilized the functionality of the web platform Roboflow [8], which provides developers with comprehensive services for building computer vision applications, including data labeling in training sets.

Due to the smaller number of data in the training set, we decided to augment the data, as shown on Figure 2. We utilized **imgaug**, which is a dedicated library for augmenting training sets using various augmentation techniques it supports [3]. To retain the entire court on the image after augmentation, we read the coordinates of the volleyball court's corners and determined the appropriate transformations based on their distance from the image edge. Transformations were also performed in random order using the Sequential function. After the transformations, we checked whether all the marked points of the court remained within the image window; if any point was outside, we repeated the process up to five times and, in case of failure, applied horizontal flipping.

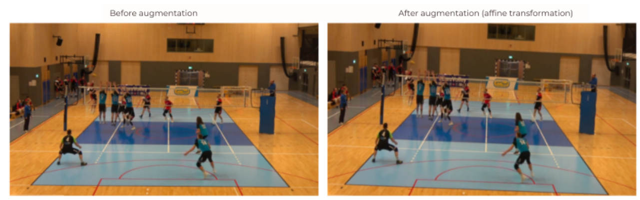

Figure 2: Original image (left) and rotated augmented image (right).

4.2 Perspective transformation

Perspective transformations have been applied in various fields, including autonomous driving, where footage from multiple cameras mounted on a vehicle was reshaped using perspective transformation to return a bird's-eye view covering the entire surroundings of the car, making it easier to assess distances between objects around the vehicle [7]. In our case, we wanted to create a bird's-eye view, that is, a top-down perspective of the volleyball court, to facilitate easier subsequent game analysis. Using a library specialized in computer vision called OpenCV, we first calculated the homography matrix, which was then used to transform the original perspective into a top-down view, as shown on Figure 3

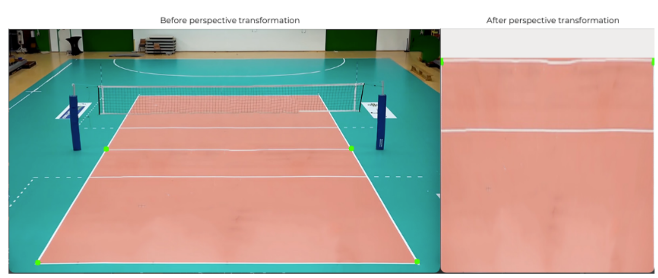

Figure 3: Original image with court edges marked by green dots (left) and image after perspective transformation (right).

4.3 Implementation and training of CNN models

For training, we chose two separate convolutional neural network models: the segmentation neural network U-Net, which was used for detecting the volleyball court, and the YOLOv8 model, which was used for detecting the ball. In both cases, we followed an approach where separate datasets were used for training and testing, meaning none of the images used during training were later used for testing the performance of any of the neural network models. The segmentation neural network U-Net, which has a symmetrical structure provided by the encoder and decoder parts, was implemented using the open-source machine learning framework PyTorch.

For the YOLO model, we decided to use one of the newer versions, specifically version eight, developed by Ultralytics [9]. Although we could have conducted training on Ultralytic's platform, we preferred to install the Ultralytics library locally and integrate it into our framework. For training purposes, we used a pre-trained YOLOv8 model (which we further fine-tuned with our own data using previously labeled data, stored in the YAML format (a data serialization format), which was generated on the Roboflow platform during data annotation).

4.4 Web application

To ensure better accessibility, regardless of the location and device of the end user, a web application was developed. The main purpose of this application is to use trained neural network models for detecting the volleyball court and tracking the volleyball through image frames during a short segment of a volleyball match. For development we used the Flask web microframework for the backend of the application, while the interface was enhanced and improved using the open-source CSS framework Bootstrap.

5 RESULTS

The final solution is essentially divided into two main parts. In the first part, the user can choose between previously prepared clips or select their own video from the device through which they are accessing the web application. After selecting the video, the user can start the analysis by clicking a button, which is divided into four key components, as shown in Figure 4:

- Detection of the volleyball court using the trained CNN U-Net model,
- Tracking the volleyball and detecting it using the CNN model YOLOv8,
- Perspective transformation of the volleyball court,
- Attack direction, where the movement of the volleyball is shown through a sequence of image frames.

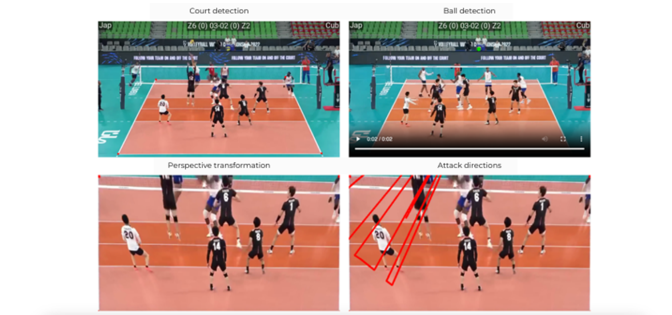

Figure 4: Web application steps: top left - Court detection with U-Net, top right - Ball tracking with YOLOv8, bottom left - Perspective transformation, bottom right - Ball trajectory across frames.

Despite the successful implementation of the desired functionalities, the application does not perform perfectly in specific situations. This becomes evident mainly in cases where the input video contains image frames different from those used to train the CNN. The most common issues we observed were:

- Different camera positions: when the volleyball match is recorded from a different camera angle than those used in the training dataset.
- Multiple lines on the court: when the video includes multiple lines that do not pertain solely to the volleyball court.
- Color scheme of the ball: in some leagues (even in top leagues, for example in Italy), they are using a ball of different color. Additionally, color schemes of the court might overlap with the ball, making it challenging to detect the ball accurately.
- Key court points or ball covered by players: when key points of the court are covered by players, as shown in the Figure 5, or by other obstacles (e.g., the net covering the line on the court farthest from the camera if the camera is positioned too low), resulting in difficulties in detecting the volleyball court or the ball.
- Real time processing limitations: based on computational resources and web application complexity, real-time streaming and analysis of video (especially with higher resolution) might introduce latency or affect performance.

Figure 5: Key court points covered by players.

6 CONCLUSIONS

In this paper, we presented the process from the initial idea to a functional web application that provides a solution for the original concept, which was the automatic analysis and visual presentation of the results to the user. The methods used were based on collecting, preparing, labeling, and augmenting data, which were then used to train two CNN. The trained models were then used, in conjunction with the perspective transformation of the court, to analyze and visually present the results to the user. The final result was presented as a user-friendly web application, where the user can select a desired video and receive a basic analysis within seconds, including court detection and volleyball tracking.

However, despite a successful implementation, the application has some weaknesses and scenarios where the results are not as expected. Challenges include different camera positions, diverse ball color schemes or multiple balls at the same time, interference that may obscure key court points or the ball itself. Additionally, real-time processing can be affected by computational limitations and video quality, potentially impacting performance.

Nonetheless, the implemented application serves as a foundation that allows for numerous upgrades, which could be inspired by existing solutions and improve upon their shortcomings. As a final result, we could provide users with real-time statistics of a volleyball match, and the platform could be expanded to include other sports, thereby attracting a wider range of users. All these reasons encourage the realization that the integration of modern technologies into all segments of our lives, including sports, is no longer a binary question but merely a matter of time.

REFERENCES

[1] Avais. 2023. Our Features. https://www.avais.ai/features. Accessed: 2024-06-02.

[2] Balltime Academy. 2024. What is Volleyball AI. https://academy.balltime.com/getting-started/what-is-volleyball-ai. Accessed: 2024-06-02.

[3] Imgaug. 2020. Documentation. https://imgaug.readthedocs.io/en/latest/. Accessed: 2024-06-02.

[4] Inside FIFA. 2022. Semi-automated Offside Technology to be Used at FIFA World Cup 2022™. https://inside.fifa.com/technical/media-releases/semi-automated-offside-technology-to-be-used-at-fifa-world-cup-2022-tm. Accessed: 2024-06-02.

[5] B.T. Naik, M.F. Hashmi, and N.D. Bokde. 2022. A Comprehensive Review of Computer Vision in Sports: Open Issues, Future Trends and Research Directions. *Applied Sciences* 12, 9 (2022), 4429.

[6] Olympics. 2024. IOC Takes the Lead for the Olympic Movement and Launches Olympic AI Agenda. https://olympics.com/ioc/news/ioc-takes-the-lead-for-the-olympic-movement-and-launches-olympic-ai-agenda. Accessed: 2024-06-02.

[7] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. 2016. You Only Look Once: Unified, Real-Time Object Detection. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 779–788.

[8] Roboflow. 2024. Our Company. https://roboflow.com/about. Accessed: 2024-06-02.

[9] Ultralytics. 2024. YOLOv8 Models Documentation.

[10] Wikipedia. 2024. Hawk-Eye. https://en.wikipedia.org/wiki/Hawk-Eye. Accessed: 2024-06-02.

[11] World Health Organization. 2024. Sports and Health Initiative. https://www.who.int/initiatives/sports-and-health. Accessed: 2024-08-30.