The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XLVIII-2/W3-2023
ISPRS Intl. Workshop “Photogrammetric and computer vision techniques for environmental and infraStructure monitoring, Biometrics and Biomedicine”
PSBB23, 24–26 April 2023, Moscow, Russia

# METHOD OF SPATIOTEMPORAL LOCALIZATION AND CLASSIFICATION OF EVENTS ON A VIDEO OF A SPORTS GAME

G. Afanasev[1], L. Mestetskiy[1], D. Poimanov [1], S. Serov[1]*

[1] Lomonosov Moscow State University, Faculty of Computational Mathematics and Cybernetics, 119991 Moscow - afanasyev-8g@yandex.ru, mestlm@mail.ru, dima.poimanow@yandex.ru, sergey.s.serov@gmail.com

**Commission II, WG II/8**

**KEY WORDS:** Spatiotemporal Localization, Pose Recognition, Trajectory Detection, Composite Bezier Curve, Video Analysis.

**ABSTRACT:**

The paper considers the problem of identifying game events in a video game of beach volleyball. As a method for solving the problem, a combined approach is proposed based on recognizing the trajectory of the ball and constructing skeletal representations of the players. Temporal localization of game events is carried out by searching for the changes in the ball trajectory using the construction of a composite Bezier curve. Spatial localization is performed by the positioning of the ball and the players in the frame at the moment of the trajectory change. The classification is based on the skeletal representation of the player touching the ball. The experiments performed show that the algorithm is able to perform spatiotemporal localization with high precision and recall.

### 1.1 Introduction

With the recent development of computing and video recording technologies, there are more and more video sports games. Analysis of the game actions of athletes is extremely important to achieve better results. However, manually sequentially searching for certain game events on long video recordings of matches takes a lot of time for players and coaches and does not provide easy navigation through the found fragments. In this regard, the task of developing an interactive analytical system that would accept a video recording of a sports game match as an input and provide the coach working with it with the ability to quickly search for a given type of game actions and direct access to video fragments containing such actions becomes relevant.

More formally, this problem can be reformulated as a problem of space-time localization and classification o f g ame events. In this paper, an attempt was made to solve the problem of spatiotemporal localization and classification o f g ame events on video recordings of beach volleyball matches made from a static camera.

To date, a free universal solution to this problem does not exist. The software products offered on the market are highly specific and not available for general use.

### 1.2 Problem statement

Let us introduce the necessary concepts.

Let’s call a game event a video fragment that demonstrates the interaction of a player with a volleyball in one of four ways: an underarm pass, an overhand pass, an attack, a block.

Let’s call the player’s skeleton $K$ the set $(k_1, k_2, ..., k_{14})$ of points on the image, connected by segments in a certain order, demonstrating the human pose.

Let’s call the ball’s trajectory change a trajectory point at which it abruptly changes its direction due to interaction with the player.

Let a video fragment of a beach volleyball game consisting of $N$ frames, shot with a static camera, be given, which may contain one or more game events. Let the classes $\{C_i\}_{i=1}^{4}$, of game events be also given, as well as the class $C_0$ denoting the absence of a game event.

It is necessary according to the presented video fragment:

1. Determine the trajectory $\{(x_t, y_t)\}_{t=1}^{N}$ of the ball in the frame and select on it all times $\{n_i\}$, at which the trajectory has changes.
2. Analyze all game events on frames with the ball trajectory changes and attribute each of them to one of the $\{C_i\}_{i=0}^{5}$ classes.

### 1.3 The proposed algorithm

If the player interacts with the ball, then the ball’s flight path contains a break point.

The solution of the set task consists of several stages, shown in the general scheme (Fig. 1):

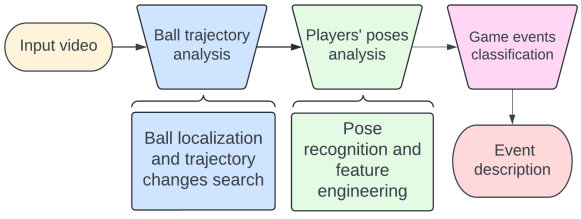


Figure 1. General scheme of the method

* Corresponding author



This contribution has been peer-reviewed.
https://doi.org/10.5194/isprs-archives-XLVIII-2-W3-2023-1-2023 | © Author(s) 2023. CC BY 4.0 License.

1. Ball trajectory analysis. For each image from the input sequence, the ball is detected. The obtained coordinates are combined into one common trajectory. The search for break points characterizing the moment of interaction with the ball is carried out.

2. Analysis of player poses. For all frames in the time neighborhood of the trajectory changes, the skeletal representations of the players are calculated. Based on the available coordinates of the ball and the skeletons of athletes, one player is determined to carry out the interaction.

3. Event Classification. For the selected player, a feature vector is constructed, consisting of the coordinates of the points of the skeleton. Additionally, the angles between body parts are calculated, specifying the athlete’s posture, speed of movement and direction. Based on the collected data, a classification is carried out.

Below we describe these steps in more detail.

## 1.4 Ball trajectory analysis

**1.4.1 Ball localization** At the localization stage, the problem of finding the coordinates of the ball on each frame of the video sequence is solved. In this work, the localization problem has its own specifics due to the high speed of movement and the small size of the desired object.

In this article, the localization problem is solved on the basis of the TrackNet neural network model (Huang et al., 2019) (Fig. 2), which combines approaches that, on the one hand, analyze more than one frame on a video simultaneously, and, on the other hand, do not require the obligatory presence of a history of the movement of the ball.

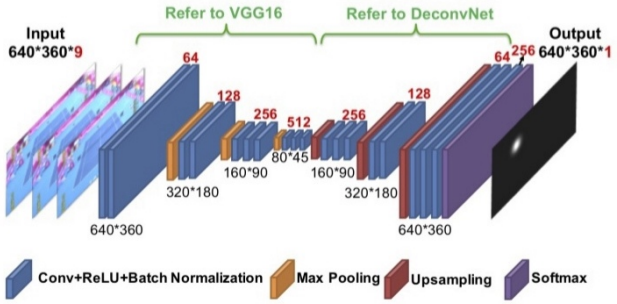


Figure 2. TrackNet network architecture

The network takes 3 consecutive frames from the video as input, applies an encoder taken from the VGG16 network (Simonyan and Zisserman, 2015) to them, and then receives the result using the DeconvNet decoder (Noh et al., 2015). The output of this network is a grayscale image, which displays the probabilities of finding the center of the ball in each pixel.

The original neural network was pretrained on tennis footage, while beach volleyball footage was used for the experiments in this paper. Therefore, as part of this work, it was further trained on the images of the game of beach volleyball, marked up by the authors themselves. However, it is impossible to label such a quantity for the case of playing volleyball manually, not only because of the large labor costs, but also due to the lack of a large amount of data for labeling. Therefore, only the last 5 layers of the decoder were retrained. The training was carried out on 500 training frames for 50 epochs of 200 steps.

**1.4.2 Trajectory smoothing** At the previous stage, a time series of ball coordinates was obtained, which has a number of disadvantages. Firstly, it contains outliers - incorrect ball locations that are farther from the rest of the found points of the trajectory beyond the allowable threshold. In addition, the trajectory is noisy due to the fact that the neural network recognizes the center of the ball with errors. For a more accurate analysis of the trajectory, it is necessary to smooth it, that is, remove all noise breaks without affecting the general appearance of the trajectory.

Since the ball periodically flies out of the frame on the video recordings used, we will divide the time series into segments so that there is only one ball inside each segment and is constantly in play. The Euclidean distance between two adjacent points of one segment must not exceed the experimental threshold value - otherwise the segment at this point is divided into two. The result of the network operation after removing outliers is shown in Fig. 3.


Figure 3. Trajectory after removing outliers for a 2 minute video. Different segments of the trajectory are colored differently

Further, for clarity, we will use only one segment for demonstration (Fig. 4).


Figure 4. One segment of the trajectory

In this example, you can see that the trajectories are still quite noisy, since the network detects the ball up to the size of the ball. For analysis, such a trajectory needs to be smoothed. By smoothing we mean the process of bringing a noisy trajectory, represented as a set of time-ordered points, to a visually smoother form.

As a smoothing algorithm, we will use non-linear smoothing with Bezier curves. The paper makes an assumption that if there is a parabolic ball trajectory represented by a sufficiently large set of noisy points, then the Bezier curve constructed from these

points will approximate this trajectory with high accuracy. Here we use higher-order Bezier curves, since the perspective projection of the ballistic trajectory of the ball onto the frame plane has an order higher than the second.

Let’s build a Bezier curve for the example trajectory. The set of trajectory points is used as the characteristic polygon.

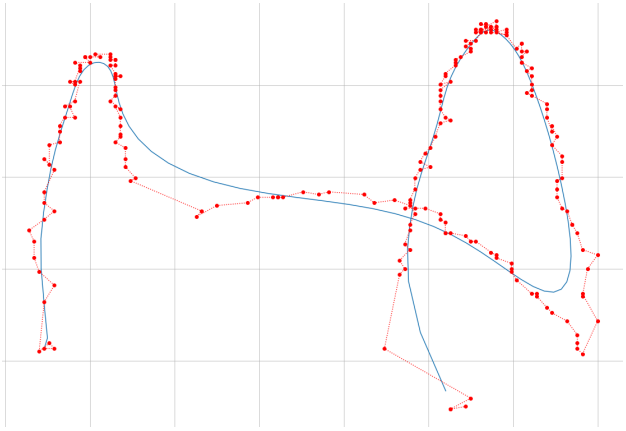

Figure 5. Bezier curve built from the points of the path. The red dots are the original path, the blue line is the Bezier curve built from them

It can be seen that the curve approximates the trajectory poorly due to the fact that in places where the trajectory changes its direction, the approximation is with low accuracy. In such a situation, the logical solution would be to approximate the trajectory using a compound Bezier curve. Thus, the trajectory smoothing algorithm is implemented as an iterative process:

1. Initialization of the list of point groups, which at step 0 will consist of one element containing all points.
2. Approximation of all groups of points by Bezier curves.
3. Finding the point furthest from the resulting Bezier curves and dividing the group of points where such a point was found into 2 groups: before it and after it.
4. Jump to step 2 if the exit condition is not met.

As an exit condition, this paper uses a comparison with the threshold of the distance from the point at which the curve is divided into two, to the Bezier curve itself. If the distance from such a point to the curve is less than the threshold, then the distance of all other points to the curve lies within these limits.

When applying this smoothing method, the ball’s flight path will be approximated by a compound Bezier curve. The connection points of the parts of this curve are the desired changes of the trajectory. The result of the algorithm for the demo curve is shown in Figs. 6, 7. The Fig. 7 shows that during the analysis noise trajectories can be formed, which are removed from consideration using a lower limit on the length of the curve.

### 1.5 Players’ poses analysis

The next stage of the proposed method is the analysis of athletes’ postures. The input is the result of the work of the previous algorithm for analyzing the ball trajectory, namely the frame numbers corresponding to the trajectory changes, and the coordinates of the ball center. It is necessary to describe the event on this frame using a numerical feature vector for its classification.

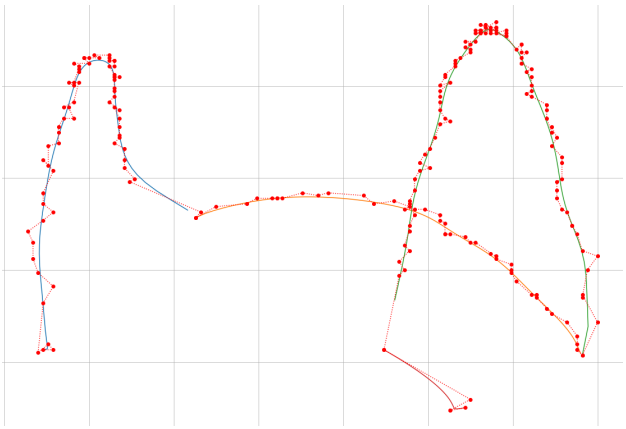

Figure 6. The result of the algorithm for the example trajectory. The dots represent the original path, and the line represents the Bezier curve

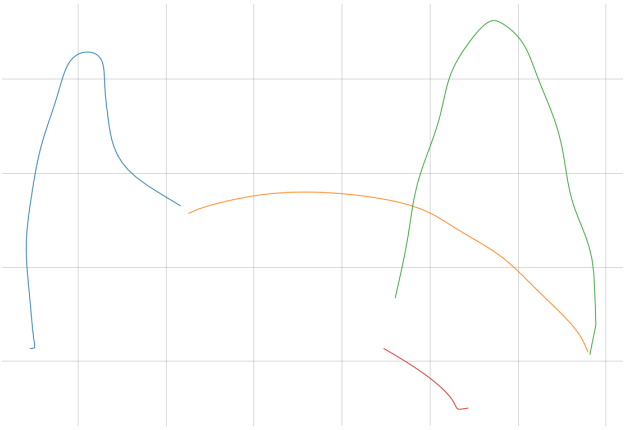

Figure 7. The result of running the algorithm for the sample trajectory without rendering the original points. You can see small noise Bezier curves at the bottom

**1.5.1 Building skeletal player representations** First, we construct skeletal representations of the players. The skeletal representation is a set of 15 points (14 points are used if the face could not be recognized) connected in accordance with the human skeleton (Figs. 8, 9). To do this, we use the OpenPose neural network method (Cao et al., 2021), which, based on the input image, builds a list of skeletal representations of people present in the frame.

The OpenPose algorithm is applied as follows: for a known frame number containing a ball trajectory change, the skeletal representations of the players are constructed on this and adjacent frames.

**1.5.2 Skeletal representations postprocessing** After applying the OpenPose algorithm, the resulting skeletons must be analyzed.

1. First, sometimes the skeletal representations of the players may not be detected due to the fast movement of the players and the complex image background (Fig. 8). Therefore, for frames in which the athlete’s skeleton could not be found, the skeleton is copied from adjacent frames. This is acceptable due to the high frame rate (30 frames / sec) - the athlete is not physically able to move far beyond the time period of 1-2 frames.

This contribution has been peer-reviewed.
https://doi.org/10.5194/isprs-archives-XLVIII-2-W3-2023-1-2023 | © Author(s) 2023. CC BY 4.0 License.

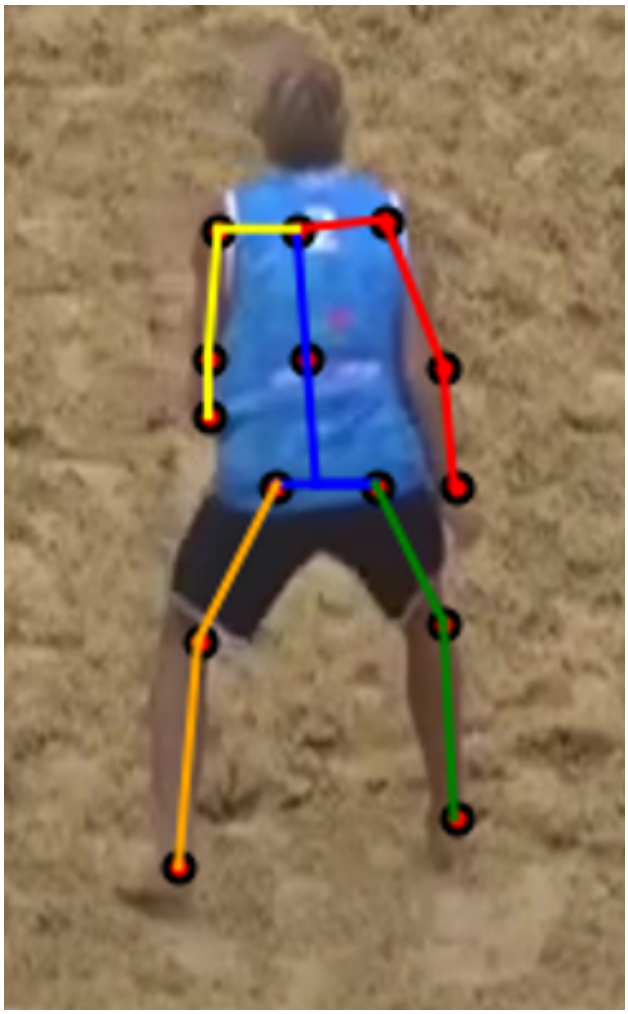

Figure 8. Player Skeleton Example

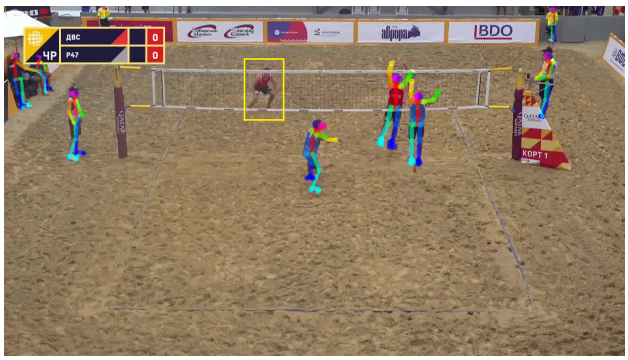


Figure 9. An example of a player whose skeletal representation failed

2. Second, we need to bind the discovered skeleton representations to the players for the frame sequence. To do this, a frame is selected on which all players (4 people) are present and selected, and all skeletal representations on it are numbered. Then, on all frames, the skeletons are sequentially assigned the numbers of the skeletons closest to them on the previous processed frame. In case of ambiguity in determining the nearest skeleton, an additional assessment of the player’s orientation is carried out. Orientation refers to the position of the left and right hand relative to the camera. At the beginning of the game, one team is facing the camera, and the other is vice versa. This characteristic of the skeleton makes it possible to distinguish athletes among themselves and unambiguously separate the selected skeletons.

As a result of the post-processing of the skeletons, on each input image, for each player, his identifier number is fixed with the corresponding skeleton.

**1.5.3 Feature engineering** The main task of the analysis of the poses of the players is to extract numerical features from the video fragment of the game event for subsequent classification. In this paper, the following method is proposed:

1. Based on the known coordinates of the ball, the player closest to it is found. The distance function is the Euclidean distance between the center of the ball and the nearest point in the player’s skeletal representation. These calculations are carried out for a frame containing a ball trajectory change. The $x, y$ coordinates of each of the 14 points of the found athlete’s skeleton are determined as components of the feature vector for classification.

2. The coordinates of the players’ skeletal representations are normalized by subtracting the minimum coordinates $x$, $y$ of the skeleton points from the coordinates of all its points.

3. In addition, the angles between body parts are calculated, since visual analysis shows that such features distinguish the desired classes. In total, 14 different angles are calculated, the arccosine of the scalar product of vectors is used for calculation. The obtained values describe the spatial location of the athlete, which is necessary to achieve the best accuracy, but is not available with only a two-dimensional image from the camera.

4. For each point of the skeleton, the rate of change of its coordinates in time is calculated, provided that the temporal frame rate is constant. Velocity is calculated as the average displacement per unit of time calculated for 10 previous frames. Thus, the feature vector expands by another 14 components.

5. Since the camera is static, for each point of the skeleton it is possible to calculate the player’s movement vector for the previous 10 frames. Both the displacements of the skeleton points between frames and the average value for the entire time interval are calculated. The vector coordinates are normalized to the vector norm and are also treated as numeric features.

Thus, in this section, an algorithm for analyzing the posture of athletes is proposed, which allows for a certain frame from a video sequence to obtain a vector of numerical features characterizing the position of the player, his speed at the time of interaction with the ball and posture. This information is fed to the input of the classifier.

## 1.6 Game events classification

The final stage of the method is the classification of the game event. In accordance with the task, 4 classes are distinguished: ”underarm pass”, ”overhand pass”, ”attack” and ”block”.

The vector of features for each event has a large dimension, so the analysis of the feature space is rather complicated. A small marked-up test video was used to reduce the dimension of the feature space in two ways: principal component analysis (PCA) and using latent semantic analysis (Truncated SVD). Visualization is shown in Figs. 10, 11.

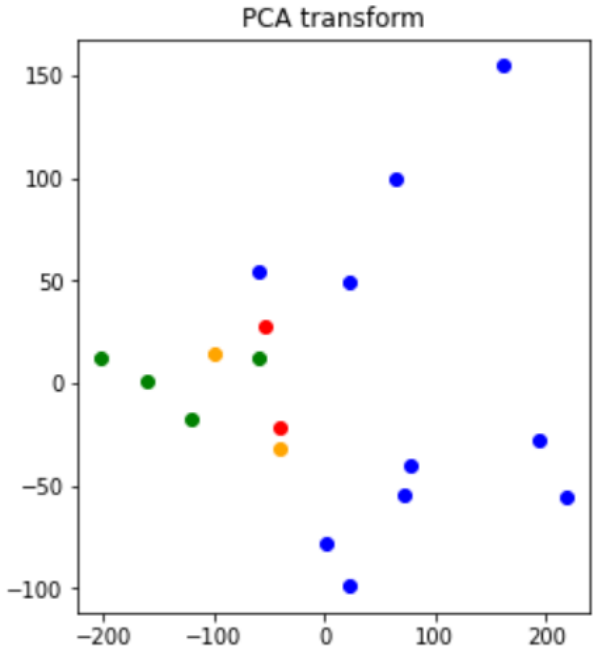


Figure 10. PCA Transform. Visualization of a reduced feature space. 4 classes are highlighted in color: blue - underarm pass, green - overhand pass, yellow - attack, red - block

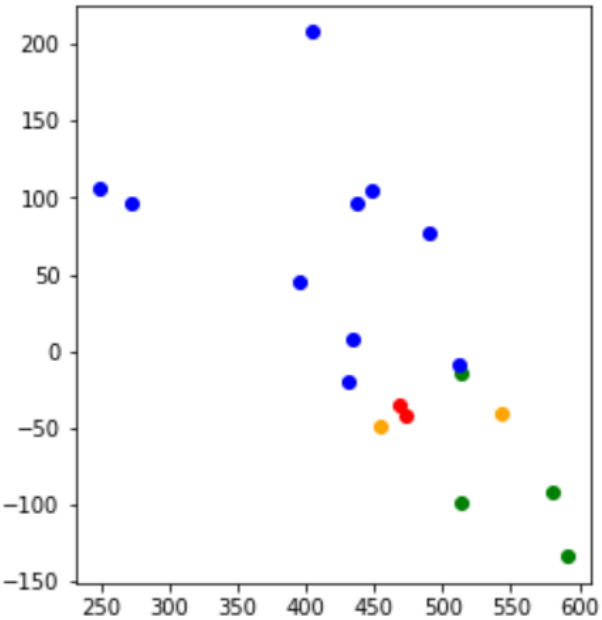


Figure 11. Truncated SVD

From the obtained images, we can conclude that the ”underarm pass” class is very different from the others, which makes it possible to classify most objects using linear algorithms. Therefore, in this paper, it was decided to divide the classification algorithm into two stages: binary classification to select the ”underarm pass” class and three-class classification to separate the remaining objects.

Both stages of the algorithm use random forest models. The final result can be seen in the Fig. 12.

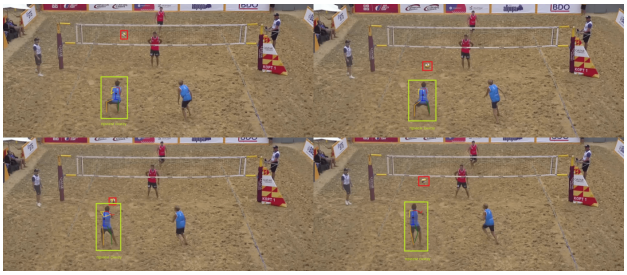

Figure 12. Visualization of the algorithm for classifying a game event

### 1.7 Experiments

Now we give examples of the operation of algorithms and the results of numerical experiments.

**1.7.1 Datasets** The algorithm was tested on two sets of videos of beach volleyball games:

- set of videos of the Moscow beach volleyball tournament. In these videos, shooting is done from the side of one of the teams from the narrow side, the camera is located from above at an angle to the field.
- set of friendly matches videos. In these videos, the video camera is positioned differently: on the right at an angle of 45 degrees to the grid from the side of one of the teams.

From the first set of video recordings, segments 10-15 seconds long were selected and marked up, showing the teams playing one point: serving, receiving the ball, the interaction of players within the team and the completion of the attack. A total of 50 gaming events were marked up. For the final evaluation of the method, two 15-second fragments from the first set of videos and a 2-minute video from the second set were used.

**1.7.2 Trajectory changes detection** In the fragments from the first set, there were 13 hits on the ball in total. There are 28 beats on the video from the second set. The algorithm of detecting the trajectory changes shows a recall of 90% and precision of 73%. Below are the frames (Fig. 13) in which a trajectory change was revealed. The green frame highlights the players who provoked a change in the trajectory.

The errors occurred in the following cases:

1. The ball after a weak hit continues to fly in the original direction.
2. The ball bounces off the ground.
3. There are several balls in the frame.
4. The ball goes out of the frame, and the trajectory is partly lost.

The most common mistake in the second video is that there is no rally in the field being filmed, but it is in the adjacent court, causing the net to trigger on the ball in the adjacent court and mark events there.

Figure 13. Event frames

Figure 14. Frames with false positives of the event search algorithm due to multiple balls and bouncing off the ground

Figure 15. The rally being played in the adjacent court

**1.7.3 Classification accuracy** The Leave-One-Out-Cross-Validation (LOOCV) approach was used to train and test the classification method at both stages due to the small dataset size. The classification algorithm was applied to the frames with events detected after the first stage and to the frames which were labeled manually. The results are shown as error matrices in Figs. 16 and 17.

The analysis of the errors showed that on the training dataset separating the class ”underarm pass” made it possible to achieve an accuracy of determining this class of 92%. Other events are closer to each other and are classified worse. The overall accuracy of recognition of the events ”overhand pass”, ”attack” and ”block” was 62%.

In the case of test videos, it was suggested that each break point is the interaction of the player with the ball. The analysis of such events showed that the most popular event was the ”underarm pass”. The accuracy of determining this class was 78%. The overall classification accuracy was 73%.

## 2. CONCLUSIONS

Thus, this paper proposes a method for spatiotemporal localization and classification of the events in a video of a sports

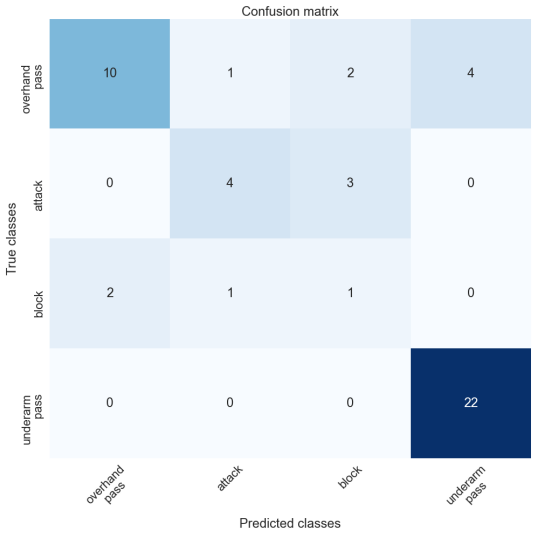


Figure 16. Train confusion matrix

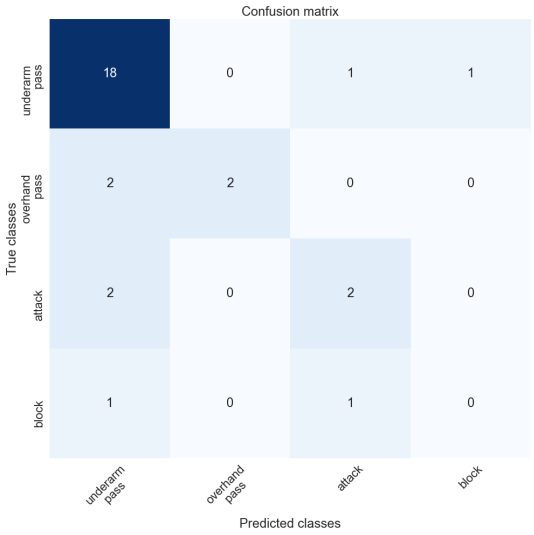


Figure 17. Test confusion matrix

game based on movement trajectories and skeletal representations. For the temporal localization of a game event, the key element is the detection of changes in the ball’s trajectory, which is performed using the algorithm of iterative construction of a composite Bezier curve from a sequence of points of the ball’s supposed centers obtained from a recognizing neural network. For the spatial localization of an event and its classification, the construction of skeletal representations of players, the construction of a vector of additional features, and a two-step classification using decision forest algorithms are used.

## REFERENCES

Cao, Z., Hidalgo, G., Simon, T., Wei, S., Sheikh, Y., 2021. OpenPose: Realtime Multi-Person 2D Pose Estimation Using Part Affinity Fields. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 43(01), 172-186.

Huang, Y.-C., Liao, I.-N., Chen, C.-H., İk, T.-U., Peng, W.-C., 2019. Tracknet: A deep learning network for tracking high-speed and tiny objects in sports applications. *2019 16th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, IEEE, 1–8.

Noh, H., Hong, S., Han, B., 2015. Learning deconvolution network for semantic segmentation.

Simonyan, K., Zisserman, A., 2015. Very deep convolutional networks for large-scale image recognition.