Object Detection in Sports Videos

M. Burić, M. Pobar, M. Ivašić-Kos
University of Rijeka/Department of Informatics, Rijeka, Croatia
matija.buric@hep.hr, marinai@inf.uniri.hr, mpobar@inf.uniri.hr

***Abstract* - Object detection is commonly used in many computer vision applications. In our case, we need to apply the object detector as a prerequisite for action recognition in handball scenes. Object detection, to be successful for this task, should be as accurate as possible and should be able to deal with a different number of objects of various sizes, partially occluded, with bad illumination and deal with cluttered scenes. The aim of this paper is to provide an overview of the current state-of-the-art detection methods that rely on convolutional neural networks (CNNs) and test their performance on custom video sports materials acquired during handball training and matches. The comparison of the detector performance in different conditions will be given and discussed.**

Keywords – object detectors; sports scenes; Mixture of Gaussians; YOLO; Mask R-CNN

I. INTRODUCTION

Object detection is one of the fundamental tasks in computer vision, with the aim to find instances of real-world objects such as people, cars, faces etc. in images or videos. Detection of an object implies prediction of the location of the object in an image along with the class it belongs to, so the challenge is to solve both object classification and object location problems.

Object detection is commonly used in many applications of computer vision such as image retrieval, security and surveillance, autonomous car driving, and many industrial applications but a single best approach to face that problem doesn't exist. The choice of the right object detection method depends on the problem that needs to be solved and on the set-up of the experiment.

In our case, we need to apply the object detector as a prerequisite for action recognition on handball scenes. For action recognition in team sports such as handball to be successful, object detection should be as accurate as possible, with reliable detection of relevant players, the ball and of other objects of interest [1]. Also, it should be able to deal with challenging conditions like the variable number of objects with a wide range of possible sizes ranging from players that can cover most of the image to the objects that are far away from the observer, that are occluded or those that can be as small as few pixels yet carry a lot of information, such as a ball. The environment in which sports videos are recorded is usually a sports hall, so the background is cluttered, with challenging illumination, with a variable number of players and with other not ideal conditions. In a highly dynamic setting of the handball domain, the motion blur and shadows that players cast under artificial illumination are often present in the videos, and the shape of the actors themselves vary greatly, making the problem even harder.

There are many other factors which can degrade the detection of players, of the ball and of the lines on the playground that humans don't even notice since it comes naturally to us.

To tackle the object detection problem, many approaches have been proposed including the Viola-Jones detector with Haar Cascades [2], HOG gradient-based approaches [3], segmentation and template matching approaches, and recent state-of-the-art methods that rely on deep convolutional neural networks (CNNs). In the last few years, CNNs have achieved a tremendous increase in the accuracy of object detection and are widely considered as the de facto standard approach for the most image recognition tasks.

Object detection in videos presents additional challenges, as it is usually desirable to track the identity of various objects between frames. It can be performed applying an object detector frame by frame, similarly as in case of images, or by using some kind of multi-frame fusion. The main difference between these approaches is, in fact, a compromise among speed, accuracy and required computational power.

The rest of the paper is organized as follows: in Section II. we will present used the image and video object detectors that rely on CNN and emphasize their strengths and weakness. We have examined their performance on a custom dataset consisting of indoor and outdoor handball scenes recorded during training and matches. The comparison of the detector performance in different conditions and discussion are given in Section III. The paper ends with a conclusion and the proposal for future research.

II. OBJECT DETECTORS

Object detection and recognition includes both object classification and objects location problems. The desired result is to have a bounding box around a detected object that is labeled with its corresponding class label.

One of the most notable and widely used object detectors in the past was a specialized face detector developed by Paul Viola and Michael Jones [2]. At that time, it was most precise and very fast, being able to perform face detection in real-time on webcam feed using hand-coded Haar features and a cascade of classifiers to make a prediction.

To realize more general object detectors that can detect many object categories, as opposed to detectors tailored for a specific object class, e.g. face, a viable approach is to start with a simpler task of image classification. In image classification, researchers are nowadays mostly focused on convolutional neural networks (CNN) that are strongly

influenced by the results of authors of [3]. At first, CNNs like VGGNet [4], Inception [5], etc. were used for classification only. The process of image classification takes an image as input and gives as the output a prediction of the existence of a class or multiple classes in case of multi-label classification, but without providing location information [6].

In order to extend the application of classifiers to the problem of object localization and to be able to detect object across the entire image, a sliding window approach is suggested. In that case, windows of different sizes corresponding to expected object sizes at various scales are positioned over overlapping parts of images to isolate parts of images that can be independently processed. If the classifier happens to recognize an object inside the window it will be labeled and marked by a bounding box for future processing. After processing the whole image, the result is a set of bounding boxes and corresponding class labels. However, the result can have a large number of unnecessary overlapping predictions. Also, the simple implementation of the sliding window approach can be very time consuming. With further development and by production of much more capable hardware, CNN based algorithms have been used to detect and localize objects as well. The Region with CNN features (R-CNN) (Fig. 1) [7] was a successful method that tried to optimize the sliding window approach. Before the image would be fed to a convolutional network for feature extraction, R-CNN would first create bounding boxes, called region proposals, using a selective search process [8]. After classification of each region using support vector machines, (SVM), R-CNN performs a linear regression on region proposals with regard to the determined object class to generate tighter bounding box coordinates.

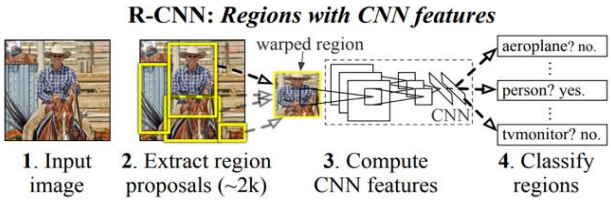

Fig. 1. Principle of Selective search inside R-CNN according to [7]

After this method was first proposed, a series of improved methods followed a similar approach. Fast R-CNN [9] brought Region of Interest Pooling (RoIPool) which reduced the number of forwarding passes and have managed to join extracting image features (CNN), classification (SVM) and bounding boxes tightening. Faster R-CNN [10] improved selective search process by reusing CNN results for region proposals instead of running a separate selective search algorithm.

A. *Mask R-CNN*

Mask R-CNN [11] is an extension of Faster R-CNN that adds a parallel branch for predicting segmentation masks on each Region of Interest (RoI), in addition to existing branches in the network that output class labels and bounding box offsets (Fig. 2). The new mask branch is a small fully connected network (FCN) applied to each RoI.

The Mask R-CNN otherwise follows the two-stage design of Faster R-CNN. The first stage is a Region Proposal Network (RPN) that proposes candidate object bounding boxes, or regions of interest (RoI). The RPN consists of a deep fully convolutional network that takes an image and outputs a feature map, upon which a smaller network is applied in a sliding window fashion. The smaller network takes a spatial window of the feature map, further reduces the feature dimension and then feeds them to two fully-connected layers, one that outputs the bounding box coordinates of proposed regions, and the other that outputs an „objectness“ score for each box, which is a measure of membership to a set of object classes vs. background. For each spatial window, k regions are proposed simultaneously based on reference boxes with pre-defined aspect ratios and scales called „anchors“, representing general object shapes, e.g. a tall box for a person.

The training data for RPNs is generated from labeled ground truth data of object boxes in images, such that positive labels are assigned to anchors with the Intersection-over-Union (IoU) overlap with a ground-truth box greater than 0.5. In this way, multiple anchors may be labeled as positive based on a single ground-truth box.

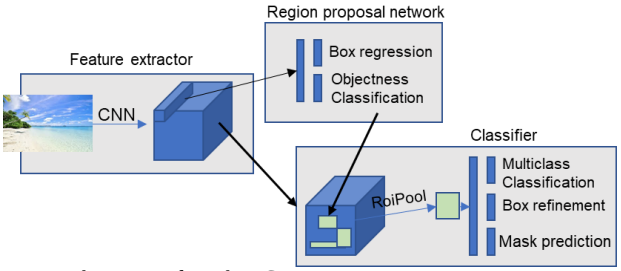

Fig. 2. The stages of Mask R-CNN

The RPNs are trained end-to-end by backpropagation and stochastic gradient descent SGD. The images are resized such that their scale (shorter edge) is 800 pixels. From each training image, N RoIs are sampled so that the ratio of positive to negatives examples is 1:3, in order to avoid the dominance of negative examples in the data. In the Mask R-CNN, the COCO dataset was used for training, while in the Faster R-CNN the PASCAL VOC was used.

Mask R-CNN generates masks and bounding boxes for all possible classes independently from classification, and finally, the result of the classification branch is used to make the selection of boxes and masks.

B. *YOLO object detector*

The “You only look once” (YOLO) [12] is another method showing promising results. It is reported to be less accurate in some cases than previously mentioned but it is much faster using the same hardware.

YOLO and its second revision YOLOv2 [13] are similar to R-CNN, in that they use potential bounding boxes from which convolutional features are extracted, but differ from the Faster R-CNN systems by using a single-stage network architecture to predict class probabilities and bounding boxes without a separate stage for the region of interest proposal.

The system divides the input image into a cell grid and produces a probability distribution of object classes for each cell. At the same time, a fixed number of candidate bounding boxes with the corresponding confidence scores

are predicted for each cell (2 in the original YOLO implementation). The confidence score measures both how confident the system is that the box contains an object, and how accurate the box is. Target confidence values for cells containing no objects is zero, while for other the confidence should be the intersection-over-union score between the predicted and the ground truth boxes.

During training, if an object spans multiple cells, only the cell containing the object center is ”responsible“ for predicting the bounding box for that object. In other words, the loss function for bounding box regression doesn't penalize all cells containing the same object, but only the central one.

Even though more than one bounding box is proposed per cell, only one class is predicted in each cell (Fig. 3). The network architecture of the original YOLO model consisted of 24 convolutional layers followed by 2 fully connected layers, where the convolutional layers extract features from the image while the fully connected layers output box predictions and probabilities.

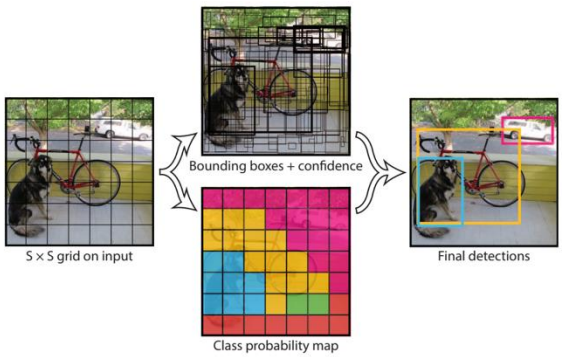

Fig. 3. YOLO object detection model divides the image into a grid and for each cell predicts bounding box, confidence for these boxes and class probability.

In the YOLOv2 system, this network architecture was replaced by a model with 19 convolutional layers with mostly 3x3 filters and 5 max-pooling layers, called darknet-19. The fully connected layers of YOLO were removed, and the bounding box proposal is modified, so instead of box coordinates, transformations of predefined anchor boxes are predicted. This is similar to the Mask R-CNN network, which also outputs shifts of pre-defined anchor boxes. In the YOLOv2 case, the anchor boxes are determined using the k-means clustering on a training set of ground truth bounding boxes and the translations of the boxes are relative to the grid cell. For each cell, 5 bounding boxes are proposed. The class predictions are now coupled with anchor boxes instead of cells, so for each bounding box, a class is predicted in addition to the objectness score, i.e. the confidence score of a box containing an object.

The network was trained on the combined dataset consisting of public available MS COCO detection dataset [14] and the top 9000 classes from the full ImageNet. The training was performed in several steps.

C. Mixture of Gaussians method

A common approach for locating moving foreground objects in scenes with predominantly static background involves background subtraction, where an approximation of background, usually an average image of several frames, is subtracted from a current frame (Fig. 4). Regions where this difference between the current frame and the background is greater than a chosen threshold are marked as foreground.

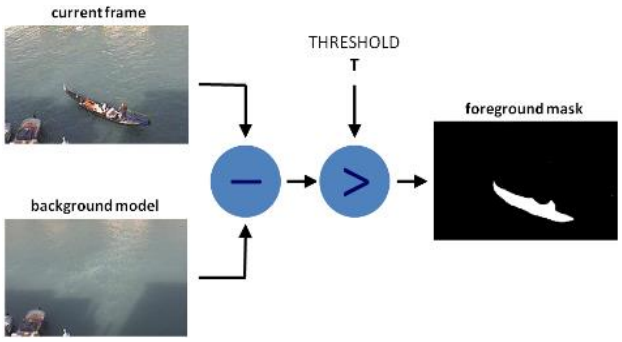

Fig. 4. The principle of background subtraction approach [15]

In practice, even in scenes shot with a stationary camera, the background is hardly ever static and can vary in time due to changes in lighting conditions, time-varying textures of background objects, e.g. waves on the water, clouds, etc. Several different background models and means of updating them have been developed to deal with this problem. Here we use the Mixture of Gaussians model (GMM) [16], where each pixel in an image is modeled as coming from a mixture of K Gaussian distributions that are continuously updated as the video progresses in time.

The GMM parameters are estimated based on the recent history of each pixel, using an online K-means approximation of the EM algorithm. Every new pixel value is checked with the existing K Gaussians to find a match, defined as a pixel value within 2.5 standard deviations of a distribution. If there is a match, the weights of distributions in the mixture are updated to give more weight to the matched distribution, and the mean and variance parameters of the matching distribution are updated to reflect the influence of the current pixel. When there is no matching Gaussian for the new pixel value, the least probable distribution in the mixture is replaced with a distribution with a low weight, whose mean is the current pixel value, and has a high initial variance.

In order to classify pixels into background or foreground classes, a heuristic rule is applied to determine which distributions in the mixture model the background. The Gaussians in the mixture are sorted in decreasing order of weight-to-standard-deviation ratio. The first B Gaussians are chosen as the background model, where B depends on a pre-set minimum portion of the data that should be considered the background.

The pixels whose values do not match one of the pixel’s backgrounds Gaussians are considered to be foreground objects and are grouped using connected components.

III. Comparison of GMM, Yolo and Mask R-CNN Detection Performance On handball Scenes

In this experiment, we have tested the YOLOv2 and Mask R-CNN object detectors based on CNNs, as well as the MOG background subtraction method on characteristic examples from the handball video domain.

The comparison was made on a custom dataset consisting of indoor and outdoor sports footage during practice and competition. The dataset contains 751 videos with 1920x1080 resolution at 30 frames per second, and the total duration of the recorded material is 1990 s. The scenes were captured using stationary GoPro cameras from different angles and in different lighting conditions. The cameras in indoor scenes were mounted at a height of around 3.5 m to the left or right side of the playground. Outdoor scenes have the camera at a height of 1.5 m. Depending on the players height, position in the field, and the camera viewpoint the size of the player in the image ranges from 40 to 240 pixels.

Both YOLO and Mask R-CNN were applied using only the CPU on the same hardware inside separate virtual machines for most reliable comparison. Publicly available pre-trained models were used with their corresponding weights build on COCO dataset, with no additional training with our own dataset.

To perform tests a high-level neural networks API Keras was applied on top of an open-source machine learning framework Tensorflow with a use of Python programing language in Ubuntu Linux environment. According to [12] the first method, YOLO, performs real-time object detection at 45 frames per second on a Titan X GPU and a fast version runs at more than 150 fps.

The other method, Mask R-CNN, predicts an object mask in parallel with the recognition of bounding boxes. This adds a small computational overhead according to [11] but gives much more information about the body posture.

On our hardware, using only the CPU, it on average 18.47 seconds for Mask-RCNN to process a 1920x1080 RGB color video frame, while YOLO performed much faster, with 0.94 seconds per frame. It is important to notice that both methods resize input data, Mask R-CNN to 1024x1024 and YOLO to 608x608 therefore using higher resolution images would not contribute much to the result.

The MOG background subtraction method is used as a baseline in the comparison vs. the full object detectors. The idea is to detect moving objects (players and balls) in the otherwise static scene of a sports field, shot with a stationary camera. Differences between consecutive frames are used to distinguish foreground objects from the mostly stationary background. It should be noted that the background subtraction methods do not attempt to determine the class of the detected foreground object, but just examine if they belong to the foreground, which may be compared with the “objectness” score of the CNN methods. In the experiment, the raw results of background subtraction were post-processed using the 3x3 square opening, 15x15 square closing and with hole-filling morphological operators.

Detectors performance are compared with the ground truth and evaluated in terms of recall, precision and F1 score [17]. For the YOLO and Mask R-CNN detectors that report confidence scores, we only considered detection whose confidence is greater than 85%, to avoid a large number of false positives otherwise reported by both detectors. For example, without the confidence threshold, the Mask R-CNN detected 27 persons per image in average, while the average number of players was in fact 10, so the confidence threshold was chosen accordingly. For MOG, all detections were considered.

To count a detection as true positive, more than half of the area belonging to the player should be inside the detected bounding box. The detector efficiency depends heavily on the number and size of objects on the scene, as well as the occlusion of objects. Fig, 5. shows the results of the evaluation of the results in the case of a simple and complex scenario. A simple scenario includes fewer objects, up to 8, close to the camera. A complex scenario is when the number of objects on the scene is equal and greater than 9, away from the camera and with the occlusions.

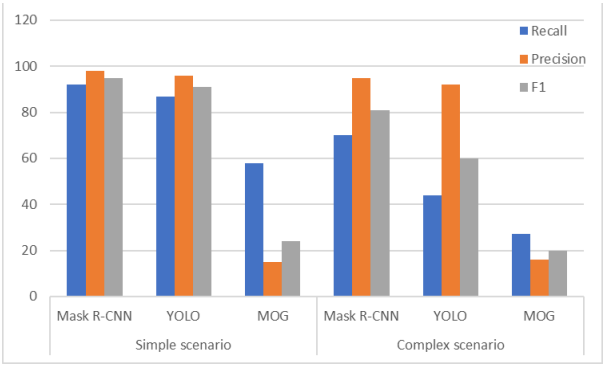

Fig. 5. Evaluation results for Mask R-CNN, Yolo and GMM in simple and complex scenarios

The detection results of three mentioned methods on characteristic handball scenes are presented in the following figures. The indoor handball scenes are presented in Fig. 6. The upper row of Fig. 6. is a result of MOG, middle of YOLO and lower of Mask R-CNN. By examining middle left figure, it is notable that compared to lower one YOLO has difficulty detecting objects smaller than 50 pixels in height.

The same is with overlapping objects which are very common in the indoor type of sports. It still made a detection of the person sitting at the far end of the court even though there are closer objects more easily distinguished to the human eye. On the other hand, MOG has detected persons that have moved, regardless of their size, but made a lot of mistakes. A lot of bounding boxers were placed where there were no players (FP), due to often highly reflective playing field, light changing and shadows that players cast under artificial illumination. Due to the different speeds of some objects, body parts such as arms and legs or head are detected as a separate object. Mask R-CNN had no problems with player detection and did not detect body parts as separate objects. Also, the players who were away from the camera, small and did not move sufficiently were not detected with MOG.

The middle picture in the second column (YOLO) shows less than a half of the objects detected by Mask R-CNN. The object of interest, the coach dressed in blue, is harder to detect compared to the rest of the video footage processed by YOLO, showing that YOLO has difficulty distinguishing objects with a similar color to the background.

Fig. 6. MOG (upper row), YOLO (middle row) and Mask R-CNN (lower row) results on indoor sport footages

MOG has no problems if an object with a similar color to the background is moving since detection depends on differences between frames. Mask R-CNN and MOG were also successful in detecting the person coming to scene from the left, but MOG had several false positives.

In an example of occlusion, on the images in the right column, YOLO has better results. It detects more persons and even the sports ball. Mask R-CNN again has FP detecting a reflection of the light on the floor as a person

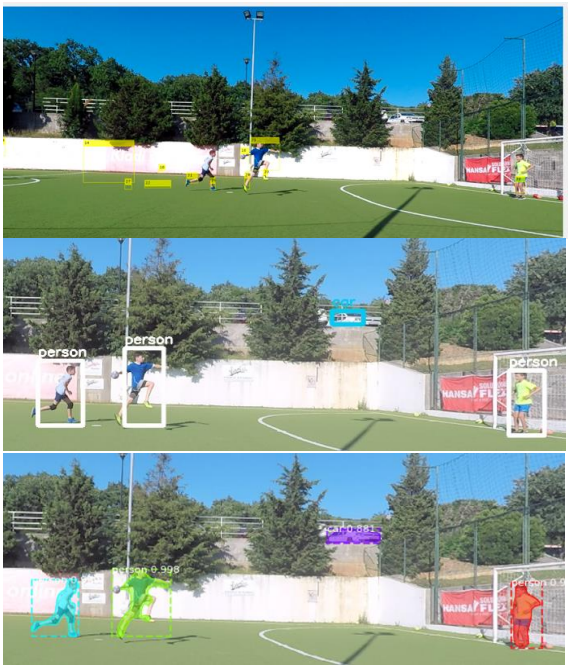

Fig. 7. Example of object detection in an outdoor scene with MOG (top), YOLO (middle) and Mask R-CNN (bottom)

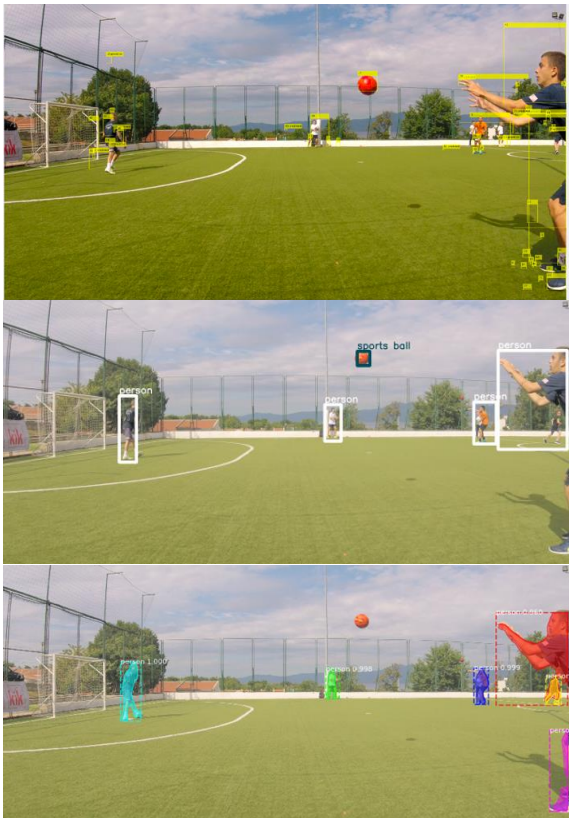

Fig. 8. MOG, YOLO and Mask R-CNN detecting a ball

(bottom left). For MOG, the net through which the scene was recorded was too big an obstacle, and as the net was shaken, MOG has detected a lot of parts of the net as foregrounds objects.

Fig. 7. shows results in an outdoor scene. The image contains few players, with no overlapping and only one partly visible object (a car) which is not important for the sport of interest. Both YOLO and Mask R-CNN have performed well, but MOG was significantly worse, with FPs and missed detections (FN).

Both YOLO and Mask R-CNN methods struggle with detecting sports balls. Figure 8. is an example where Mask R-CNN was unable to detect the ball, while YOLO and MOG detected one out of two balls. Mask R-CNN however detected one more person at a distance than YOLO. It is important to notice that shadows which can be detected as real objects have not confused any of the methods.

IV. Conclusion

The analysis of the obtained results shows that Mask R-CNN is more appropriate in the footages of team sports because it can successfully detect individual players even when they are inside a group. Also, it has more success detecting individuals further away from the camera. An additional benefit is a mask around the detected object, which, with slightly more computation power, provides significant information which can be used to isolate it from the background. The problem with this method is that it requires more time and computation power.

It was faster to test and finetune YOLO on real-time data with satisfying results. In the majority of acquired video footages, YOLO has proven to be sufficient and in case of occlusion even better than Mask R-CNN. The MOG detector works fast but has proved to have too many false detections in comparison with both YOLO and Mask R-CNN, as it was expected due to it being a binary background/foreground detector working only on motion data.

Even though Mask R-CNN lacks a possibility of the fast and easy on-the-field-analysis, taken into account further hardware and algorithm development it can be concluded with high probability this will be solved in the near future.

The one thing YOLO and Mask R-CNN methods had a problem with is the inability to reliably detect the sports ball. This could be due to fact that shape and texture of it share same features like many other objects (head, lamp, decorations). Probable solution would be to take into account existence and position of the ball near a player during the training phase to overcome this behavior. The interesting observation is that even though shadows of many objects resemble the actual object, they were (almost) never mistaken for a ground truth. Possible combination of the tested methods, which would result in benefits from both, should be further research but it falls outside the scope of this research paper.

This paper provides a promising base ground for further research of activity recognition in video material. By selecting the optimal solution and adjusting it to a certain activity it could be even possible to subtract the leading player and predict its movement or even an outcome of the action.

Acknowledgment

This research was fully supported by Croatian Science Foundation under the project IP-2016-06-8345 “Automatic recognition of actions and activities in multimedia content from the sports domain” (RAASS).

References

[1] M. Burić, M. Pobar, M. Ivašić-Kos, "An overview of action recognition in videos," *2017 MIPRO*, Opatija, 2017, pp. 1098-1103.

[2] P.A. Viola, M.J. Jones, “Rapid object detection using a boosted cascade of simple features”, in CVPR, issue 1, 2001, pp. 511–518.

[3] A. Krizhevsky, I. Sutskever and G. E. Hinton, ImageNet Classification with Deep Convolutional Neural Networks, Advances in Neural Information Processing Systems, 2012., pp. 1097-1105.

[4] K. Simonyan, A. Zisserman. “Very Deep Convolutional Networks For Large-Scale Image Recognition,“ arXiv:1409.1556., 2014.

[5] C. Szegedy *et al*., "Going deeper with convolutions," *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, 2015, pp. 1-9.

[6] M. Pobar, M. Ivašić-Kos, Multi-label Poster Classification into Genres Using Different Problem Transformation Methods; Computer Analysis of Images and Patterns, CAIP 2017, Lecture Notes in Computer Science, vol. 1042

[7] Girshick, J. Donahue, T. Darrell and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," *2014 IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, OH, 2014, pp. 580-587.

[8] J.R.R. Uijlings, K.E.A. van de Sande, T. Gevers, and A.W.M. Smeulders, „Selective Search for Object Recognition”, International Journal of computer vision, 104(2), 154-171.

[9] R. Girshick, "Fast R-CNN," *2015 IEEE International Conference on Computer Vision (ICCV)*, Santiago, 2015, pp. 1440-1448.

[10] S. Ren, K. He, R. Girshick and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 6, pp. 1137-1149, June 1, 2017.

[11] K. He, G. Gkioxari, P. Dollár and R. Girshick, "Mask R-CNN," 2017 IEEE International Conference on Computer Vision (ICCV), Venice, 2017, pp. 2980-2988.

[12] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, 2016, pp. 779-788.

[13] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, 2017, pp. 6517-6525.

[14] T. Lin et al. “Microsoft COCO: common objects in context,” European conference on computer vision, Springer, Cham, 2014, pp. 740-755.

[15] "OpenCV: How to Use Background Subtraction Methods", Docs.opencv.org, 2018. [Online]. Available: https://docs.opencv.org/3.2.0/d1/dc5/tutorial_background_subtraction.html. [Accessed: 19- Dec- 2017].

[16] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Fort Collins, CO, 1999, pp. 246-252 Vol. 2.

[17] M. Ivašić-Kos, M. Pobar, Multi-label Classification of Movie Posters into Genres with Rakel Ensemble Method; Artificial Intelligence XXXIV. SGAI 2017. Lecture Notes in Computer Science, vol 10630; Chambridge : Springer, 2017. 370-383