16th International Conference on Machine Vision Applications (MVA)
National Olympics Memorial Youth Center, Tokyo, Japan, May 27-31, 2019.

BallTrack: Football ball tracking for real-time CCTV systems

Jacek Komorowski
Warsaw University of Technology
and SAG sp. z o.o.
Chodna 51
00-867 Warsaw
j.komorowski@sagsport.com

Grzegorz Kurzejamski
Warsaw University of Technology
and SAG sp. z o.o.
Chodna 51
00-867 Warsaw
g.kurzejamski@sagsport.com

Grzegorz Sarwas
Warsaw University of Technology
and SAG sp. z o.o.
Chodna 51
00-867 Warsaw
g.sarwas@sagsport.com

Abstract

The paper describes a deep network based system specialized for ball detection in long shot videos. System comprises of flexible detector and classical particle tracking. The core contribution is incorporation of hypercolumn concept in the processing pipeline achieving real-time tracking on 12MPx videos. System achieves state-of-the-art results in ISSIA-CNR Soccer Dataset and its feasibility has been tested on 4 camera prototype system.

1 Introduction

AI-powered tracking systems are the emerging technology in sports industry. Ball tracking in particular is a core capability of any system aiming to automate analysis of the football matches or players' development. Despite numerous systems aiming at players tracking, ball trajectory estimation remains a hard task for production-class systems, as the object is textureless and visually hard to discriminate. This is even more true in case of stationary camera systems, where the field of view is usually pitch wide. Resolution of the ball can achieve less than 20 px. Optical interference and capture quality may alter visuals significantly.

The system presented in the paper is developed as a part of a computer system for football clubs and academies to track and analyze player performance during both training session and regular games. It builds up upon the work presented in [16]. We tested the ball tracking capabilities in basic scenario for ISSIA-CNR Soccer Dataset as well as an in-house dataset from system prototypes installed in academies. We present results for public databases and insights into production system challenges.

System has to overcome multiple difficulties. Due to the perspective projection, ball's size varies depending on the position on the play field. Ball's shape is not always circular. When a ball is kicked and moves at high velocity, its image becomes blurry and elliptical. Different balls used during matches have different textures and colours. Figure 1 shows few image patches with high variance in the ball appearance during a match.

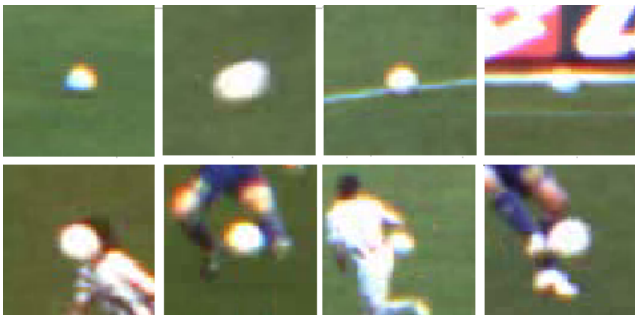

Figure 1. Exemplary patches illustrating high variance in ball appearance and difficulty of the ball detection task.

2 Related Work

The first step in the traditional ball detection methods is usually a background subtraction as in [7] or motion detection [1, 10]. The second step considers using criteria like blob size, colour and shape (circularity, eccentricity) or Circle Hough Transform as in [1, 5, 11]. A two-stage approach may be employed to achieve real-time performance and high detection accuracy as in [10]. In this scenario first step is to identify regions with high probability that the ball may be found. Then, multiple candidates are validated. [5] use multiple successive frames to improve the detection accuracy. Current state-of-the-art neural-network object detectors can be categorized as one-stage or two-stage.

Table 1. Details of *DeepBall* network architecture. Each convolutional layer is followed by BatchNorm layer and ReLU non-linearity (not showed for brevity). All convolutions use same padding and stride one (except for the first one).

Block	Layers	Output size
Conv1	Conv: 8 7x7 filters stride 2	
	Conv: 8 3x3 filters	
	Max pool: 2x2 filter	(8, 268, 480)
Conv2	Conv: 16 3x3 filters	
	Conv: 16 3x3 filters	
	Max pool: 2x2 filter	(16, 134, 240)
Conv3	Conv: 32 3x3 filters	
	Conv: 32 3x3 filters	
	Max pool: 2x2 filter	(32, 67, 120)
Conv4	Conv: 56 3x3 filters	
	Conv: 2 3x3 Filters	(2, 268, 480)
Softmax	Softmax	(2, 268, 480)

In two-stage detector, such as Faster R-CNN [12], the first stage generates a sparse set of candidate object locations. The second stage classifies each candidate location as foreground or background. One-stage detectors, RetinaNet [8] or SSD [9], do not include a region-proposal generation step. [14] uses convolutional neural networks (CNN) to localize the ball under varying environmental conditions. The limitation of this method is that it fails if more than one ball is visible. [13] presents a deep neural network classifier, consisting of convolutional feature extraction layers followed by fully connected classification layer. It is trained to classify small, rectangular image patches as ball or no-ball.

3 Deep network detector

The ball detector used in the system, called hereafter DeepBall, has been inspired by state-of-the-art detection methods such as SSD [9]. Architectures presented in these papers have been tailored for locating small objects and reducing the processing time. The core concept of the network is the hypercolumn approach introduced in [6]. Multiple anchor boxes, with different sizes and aspect ratios have been discarded as we have only one class of object with fixed shape. Detections are further used in a particle-based tracker so pixel-level bounding boxes are not needed. We evaluated network in two scenarios: single frame and three consecutive frames stacked together as an inputs.

The method takes a video frame of any resolution as an input (or alternatively three frames for times (t,-2, t-1, t)) and produces scaled down *ball confidence map* encoding probability of ball presence at each location. See Fig. 2 for an exemplary input image and corresponding *ball confidence map* computed by the trained network. Actual position of the ball is computed by choosing the maximum value or by thresholding the map in case of a training session (multiple balls used during training).

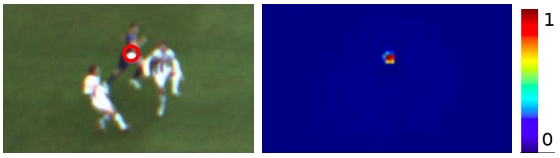

Figure 2. Part of the exemplary input frame from the test sequence with highlighted ball position (left) and corresponding *ball confidence map* (right)

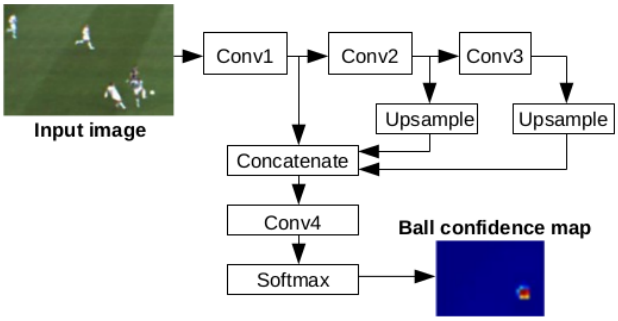

Figure 3. High-level architecture of *DeepBall* network.

The input image is processed by three convolutional blocks (Conv1, Conv2 and Conv3) producing convolutional feature maps with decreasing spatial resolution and increasing number of channels. The output from each convolutional block is concatenated and jointly fed into the final classification layer. Output of Conv1 and upsampled feature maps from Conv2 and Conv3 form the hypercolumn. The diagram depicted in Fig. 3 shows components of our ball detection network and size of outputs of each block. Conv4 is a fully convolutional classification block followed by a softmax layer.

The network output has two channels: the first one interpreted as the probability of the location belonging to the background and the other as probability of the ball. For ball detection purposes only the later map is used. Detailed architecture of each block is given in Table 1.

Concatenation of multiple convolutional feature maps from different level of the network allows analysis of bigger receptive field, meaning the context of local probability detection is multi-scaled. Conv1 has high resolution output and thus may contain information for precise localization. Conv2 and Conv3 are the coarse estimation of the local visual variations, giving

feedback on area similar to player's size. This is very important in case of players' interaction with the ball.

Loss function used for training is a modified version of the loss used in SSD [9] detector. The loss $\mathcal{L}$ optimized during the training is cross-entropy loss over ball and background class confidences:

$$\mathcal{L}(c) = \frac{1}{N}\left(-\sum_{(i,j)\in Pos}\log\left(c_{ij}^{ball}\right) - \sum_{(i,j)\in Neg}\log\left(c_{ij}^{bg}\right)\right), \quad (1)$$

where c_{ij}^{bg} is the value of the channel of the ball confidence map corresponding to the background probability at the spatial location (i, j) and c_{ij}^{ball} is the value of the channel of the ball confidence map corresponding to the ball probability at the spatial location (i, j). Pos is a set of positive examples. Neg is a set of negative examples. During training we employ hard negative mining strategy as in [9], so the ratio of negative to positive examples is at most 3:1.

The network is trained using a standard gradient descent approach with Adam optimizer. The initial learning rate is set to 0.001 and decreased by 10 after 50 epochs. The training runs for 75 epochs in total. Batch size has been set to 16.

4 Tracker

For track generation we use basic Particle Filtering approach using first-order velocity estimation in real coordinates (given calibrated camera data) with analysis history of two measure points (two frames) for particle movement. Attempts at a more advanced dynamic model were discarded as ball's trajectory is highly non-deterministic due to interactions with players. Particle weights are derived by sampling values from DeepBall confidence map used as a probabilistic measurement for each frame. This overcomes computational overhead of the filter, as each particle only samples single value from the map. Position estimation in evaluation step is performed with Mean Shift. Dispersion model is circular in a plane of the pitch and we use around 100 particles per object. Few particles from dispersion model are shown in Fig. 4. The radius of the dispersion is defined by velocity probability model of ball in the game.

Unfortunately we ISSIA dataset does not contain data for quality assessment of the tracks. Moreover we tested if adding track context could be beneficial to detection, adding capability to interpolate position for frames with no detection. We found that it is not the case, as detection algorithm is highly stable for tracks being easy to identify and highly unstable for occluded, highly changing part of tracks which produce bad interpolation data at the end. Tracks are important though in the assessment of such metrics as id switching. Such results are not included in this paper because of lack of proper public database.

Figure 4. Velocity vector and particles (yellow dots) during ball tracking.

5 Experimental results

Experimental results are presented as precision and accuracy of detections of the DeepBall module with single and multi-frame input version. It is important to note that ISSIA-CNR does not contain full tracks data and calibration data. Thus real efficiency for 3D estimation is still being prepared on suitable prototype installations. Our tests on custom proprietary CCTV installations (soon to be prepared for public) shows no significant gain in accuracy in case of ball detection when DeepBall outputs full 3D tracks. Tracks are thus needed mainly for context handling while fully occluded and when multiple balls are present.

DeepBall network is trained and evaluated using the ISSIA-CNR Soccer Dataset [2] with additional manually-annotated data from 12MPx camera CCTV systems in proprietary installations. We use standard data augmentation to increase the variety of training examples and decrease the risk of overfitting.

Table 2 contains test results: Average Precision and Accuracy of evaluated methods. We use definitions of metrics as in Pascal 2007 VOC Challenge [3]. FPS statistics are for ISSIA dataset at FullHD resolution and proprietary dataset. We achieved 24 FPS in our test with proprietary datasets, where the footage is taken from four top-quality 12MPx (4000x3000) CCTV cameras. It means real-time using Titan X class GPU. Difference in appearance for two datasets are shown in Figure 5. Thanks to incoporating NVDEC hardware decoding, CPU particle sampling and GPU AI scoring we achieved 24 FPS for whole installation of 4 cameras ona single pitch. Processing unit for that consists of two Titan X GPUs and single Ryzen 1700 CPU with 16GB of RAM.

Our method yields the best results on the test set (Sequences 5 and 6 from ISSIA-CNR Soccer Dataset). We evaluated two recent ball detection methods for comparison: [14] and [13] using the same training and test scenario.

Table 2. Ball detection method evaluation results with ISSIA groundtruth.

Method	Average Precision	Accuracy	No. of trainable parameters	FPS (ISSIA)	FPS (12MPx)
DeepBall (multi frame input)	**0.94**	**0.94**	54 226	192	23
DeepBall	**0.88**	**0.90**	48 658	192	24
DeepBall (no data augmentation)	0.792	0.899	48 658	192	24
DeepBall (no hypercolumns/context)	0.833	0.911	29 146	274	32
[14]	0.220	0.220	332 365 744	22	nd
[13]	0.834	0.917	313 922	32	nd

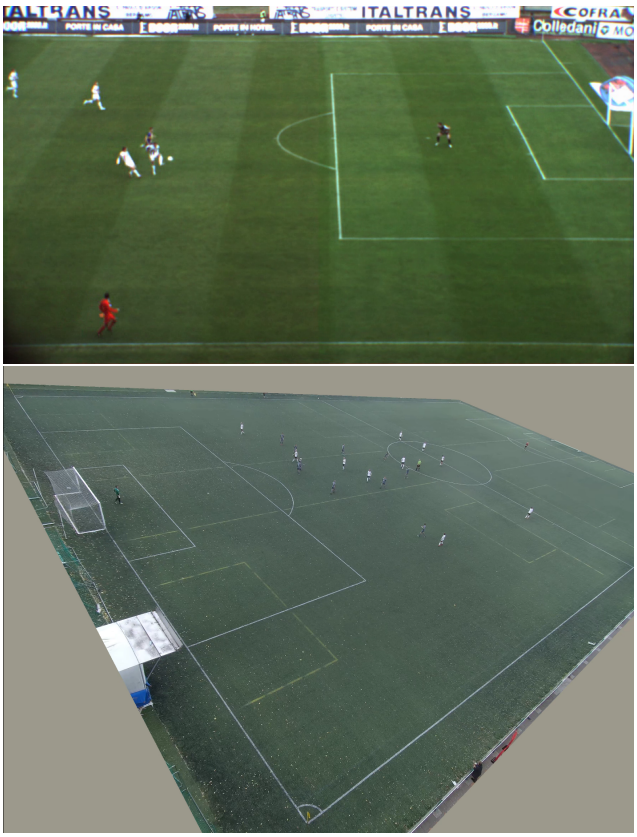

Figure 5. Exemplary frame from the ISSIA-CNR training dataset (top) and proprietary data from custom installations (bottom).

6 Production challenges and conclusions

We proposed feasible solution for ball tracking in Football games. Given very good results on public databases and real-time performance on highly demanding 12MPx images we open multiple ways to analyze football matches in production scenarios. Still, ball estimation is only one element of huge system and given efficiency can be achieved only in conjunction with GPU-accelerated decoding (CPU decoding of 12MPX is still not feasible for most current-gen processors) and wise GPU resources management. Next step for a proper ball tracking is evaluation of tracking algorithm, based on track ground-truth and 3D estimation in world coordinates, given multiple-view systems. Ground truth of such sort is now being prepared by authors for public disclosure and further system rigorous tests.

Acknowledgements

This work was co-financed by the European Union within the European Regional Development Fund grant no. POIR.01.02.00-00-0153/17-00.

References

[1] D'Orazio, T., Guaragnella, C., Leo, M., and Distante, A. (2004). A new algorithm for ball recognition using circle hough transform and neural classifier. *Pattern Recognition*, 37(3):393 – 408.

[2] D'Orazio, T., Leo, M., Mosca, N., Spagnolo, P., and Mazzeo, P. (2009). A semi-automatic system for ground truth generation of soccer video sequences. In *2009 Advanced Video and Signal Based Surveillance*, pages 559–564. IEEE.

[3] Everingham, M., Van Gool, L., Williams, C. K., Winn, J., and Zisserman, A. (2010). The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338.

[4] Felzenszwalb, P. F., Girshick, R. B., McAllester, D., and Ramanan, D. (2010). Object detection with discriminatively trained part-based models. *IEEE transactions on pattern analysis and machine intelligence*, 32(9):1627–1645.

[5] Halbinger, J. and Metzler, J. (2015). Video-based soccer ball detection in difficult situations. In Cabri, J., Pezarat Correia, P., and Barreiros, J., editors, *Sports Science Research and Technology Support*, pages 17–24, Cham. Springer International Publishing.

[6] Hariharan, B., Arbeláez, P., Girshick, R., and Malik, J. (2015). Hypercolumns for object segmentation and fine-grained localization. In *Proceedings of the IEEE*

conference on computer vision and pattern recognition, pages 447–456.

[7] Kia, M. (2016). Ball automatic detection and tracking in long shot views. *International Journal of Computer Science and Network Security (IJCSNS)*, 16(6):1.

[8] Lin, T.-Y., Goyal, P., Girshick, R., He, K., and Dollár, P. (2017). Focal loss for dense object detection. *arXiv preprint arXiv:1708.02002*.

[9] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., and Berg, A. C. (2016). Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer.

[10] Mazzeo, P. L., Leo, M., Spagnolo, P., and Nitti, M. (2012). Soccer ball detection by comparing different feature extraction methodologies. *Advances in Artificial Intelligence*, 2012:6.

[11] Poppe, C., De Bruyne, S., Verstockt, S., and Van de Walle, R. (2010). Multi-camera analysis of soccer sequences. In *Advanced Video and Signal Based Surveillance (AVSS), 2010 Seventh IEEE International Conference on*, pages 26–31. IEEE.

[12] Ren, S., He, K., Girshick, R., and Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99.

[13] Reno, V., Mosca, N., Marani, R., Nitti, M., DOrazio, T., and Stella, E. (2018). Convolutional neural networks based ball detection in tennis games. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 1758–1764.

[14] Speck, D., Barros, P., Weber, C., and Wermter, S. (2017). Ball localization for robocup soccer using convolutional neural networks. In Behnke, S., Sheh, R., Sarıel, S., and Lee, D. D., editors, *RoboCup 2016: Robot World Cup XX*, pages 19–30, Cham. Springer International Publishing.

[15] Yuen, H., Princen, J., Illingworth, J., and Kittler, J. (1990). Comparative study of hough transform methods for circle finding. *Image and vision computing*, 8(1):71–77.

[16] Komorowski J., Kurzejamski G., Sarwas G. DeepBall: Deep Neural-Network Ball Detector (2019), Proceedings of the VISAPP 2019, part of 14th International Joint Conference on Computer Vision