AI-Based Cropping of Soccer Videos for Different Social Media Representations

Mehdi Houshmand Sarkhoosh[2,3], Sayed Mohammad Majidi Dorcheh[2,3], Cise Midoglu[1,3], Saeed Shafiee Sabet[1,3], Tomas Kupka[3], Dag Johansen[4], Michael A. Riegler[1], and Pål Halvorsen[1,2,3]

[1] SimulaMet, Norway
[2] Oslo Metropolitan University, Norway
[3] Forzasys, Norway
[4] UIT The Arctic University of Norway

Abstract. The process of re-publishing soccer videos on social media often involves labor-intensive and tedious manual adjustments, particularly when altering aspect ratios while trying to maintain key visual elements. To address this issue, we have developed an AI-based automated cropping tool called SmartCrop which uses object detection, scene detection, outlier detection, and interpolation. This innovative tool is designed to identify and track important objects within the video, such as the soccer ball, and adjusts for any tracking loss. It dynamically calculates the cropping center, ensuring the most relevant parts of the video remain in the frame. Our initial assessments have shown that the tool is not only practical and efficient but also enhances accuracy in maintaining the essence of the original content. A user study confirms that our automated cropping approach significantly improves user experience compared to static methods. We aim to demonstrate the full functionality of SmartCrop, including visual output and processing times, highlighting its efficiency, support of various configurations, and effectiveness in preserving the integrity of soccer content during aspect ratio adjustments.

Keywords: AI, aspect ratio, cropping, GUI, soccer, social media, video

1 Introduction

Media consumption has expanded beyond traditional platforms, and content must be curated and prepared in various formats [32]. Today, people are spending a huge amount of time to manually generate content to be posted in various distribution channels, targeting different user groups. With the rise of artificial intelligence (AI), such processes can be automated [7, 21]. Using soccer as a case study, we have earlier researched and demonstrated tools to automatically detect events [23–25], clip events [34, 35], select thumbnails [10, 11] and summarize games [8, 9]. The next step is to prepare the content for specific platforms, ranging from large television screens to compact smartphones. Each platform and view mode might require different aspect ratios, necessitating an adaptive presentation that remains consistent in its delivery of content to audiences, irrespective of the viewing settings, i.e., cropping the original content to the target platform while preserving its essence.

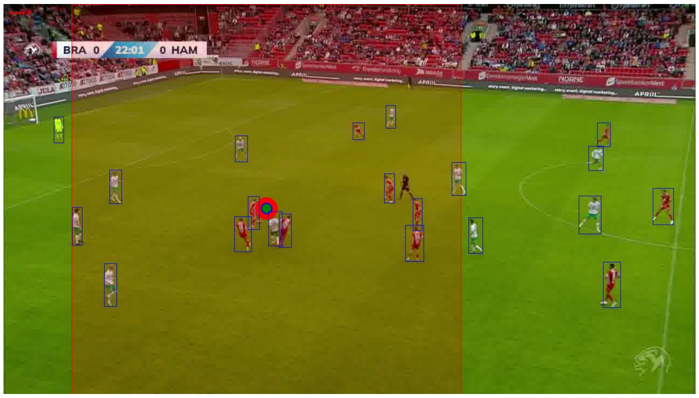


Fig. 1: Smart cropping using object detection. Red dot marks the calculated cropping-center point in the frame (the ball), red square marks the cropping area.

To capture the saliency of soccer videos, the cropping operation must consider dynamic objects such as the ball and players at the same time as the camera itself is moving. Traditionally, video cropping is performed manually using commercial software tools, selecting the area of interest in a frame-by-frame manner, which is a tedious and time-consuming operation. With the sheer volume of content and the demand for timely publishing, manual approaches are unsustainable. Hence, the domain has witnessed a shift towards automated solutions using AI.

There are several areas of research related to cropping, such as changing the video aspect ratio, e.g., content-adaptive reshaping (warping) [17, 22], segment-based exclusion (cropping) [5, 12, 19, 27], seam extraction [14, 37] and hybrids of these techniques [1, 16, 38], as well as detecting the location of important objects in a frame, e.g., single shot multibox detector [20], focal loss for dense object detection [18], faster R-CNN [29], and YOLO [28]. In the context of our work, YOLOv8 has a specialized ball detector. However, challenges inherent in soccer broadcasts, such as swift player movements, sudden changes in camera perspectives, and frequent occlusions, necessitate adjustments to YOLOv8's neural network layers. Furthermore, ensuring the consistent visibility of the soccer ball is non-trivial, and challenges such as inconsistent detection, rapid lighting changes, player occlusions, and camera angles can hinder its accurate detection.

The goal of our work is to develop an AI-based smart cropping pipeline tailored for soccer highlights to be published on social media. We have developed SmartCrop [6] for delivering various media representations using a fine-tuned version of YOLOv8 for object detection, and tracking the ball through an extended logic including outlier detection and interpolation, for calculating an appropriate cropping-center for video frames. This is shown in Figure 1. Initial results from both objective and subjective experiments show that SmartCrop increases end-user Quality of Experience (QoE). We will demonstrate our pipeline step-by-step in interactive fashion, where participants can configure settings such as the target aspect ratio, and chose among various methods for object detection, outlier detection, and interpolation.

2 SmartCrop

The guiding principle for the SmartCrop pipeline [6] is to use the soccer ball as the Point of Interest (POI), i.e., the center of the cropping area, as illustrated by Figure 1. When the ball is visible within the frame, it is the primary focal point for the cropping. If it is absent, we employ outlier detection and interpolation to select an appropriate alternative focal point. The pipeline is depicted in Figure 2.

All video formats can be processed by SmartCrop, in this demo we use an HTTP Live Streaming (HLS) playlist as **video input**. To reduce runtime, the pipeline searches inside the HLS manifest and selects the lowest quality stream available, which has the smallest resolution. After pre-processing, the first step is **scene detection**, which runs SceneDetect [3] and TransNetV2 [33]. These models jointly segment the video into distinct scenes, allowing each to serve as a separate unit for further steps.

The **object detection** module can employ one of the various supported alternative YOLOv8 models [13,36]. Here, we tested various alternatives as shown in Figure 3 (`Sc01` to `Sc05` based on YOLOv8-nano, small, medium, large, xlarge respectively), and opted for the medium architecture as a baseline after considering several trade-offs. Larger architectures offer slightly better performance, but at a significantly increased computation time. On the other hand, smaller models provide quicker processing, but suffer from reduced accuracy. Starting with `Sc06` (previously trained on 4 classes as detailed in [26]), we used the YOLOv8-medium model, fine-tuning it on a dataset that included 1,500 annotated images from Norwegian and Swedish soccer leagues, plus 250 images from a public soccer dataset [30]. The model went through multiple training scenarios with different hyperparameters, `Sc11` representing the final and most effective scenario in this series. This specific configuration involved processing high-resolution images and utilized an increased batch size for more effective learning. The training approach for featured extended training epochs, a carefully calibrated learning rate for steady model development, and a tailored dropout rate to ensure robust generalization. This structured training regimen enabled `Sc11` to achieve significant accuracy in detecting balls, players, and logos, establishing it as the optimal configuration for the object detection module.

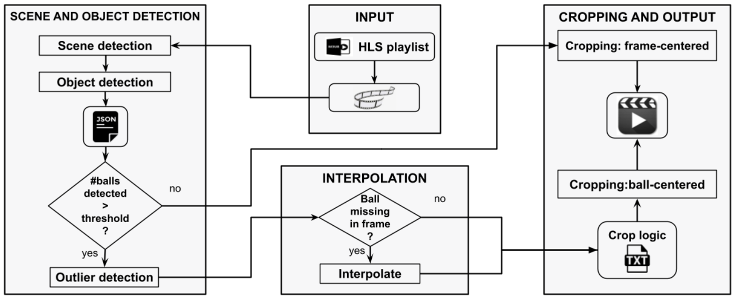


Fig. 2: SmartCrop pipeline overview.

The **outlier detection** module is designed to identify and exclude anomalous data points, known as outliers, from the detected positions of the soccer ball. This is crucial for enhancing the system's robustness. Outliers are data points that deviate significantly from the majority, lying beyond a pre-set threshold. To accurately determine these thresholds, we integrated three distinct methods: Interquartile Range (IQR), Z-score, and modified Z-score, all described in [31]. After outlier detection, we use **interpolation** to estimate the position of the ball for frames where the ball is not detected. The module can impute missing data points using various alternative interpolation techniques, including linear interpolation [2], polynomial interpolation [2], ease-in-out interpolation [15], and our own heuristic-based smoothed POI interpolation [6].

The **cropping** module is used to isolate regions of interest within the frames, reducing computational complexity and improving focus on key areas. It crops each frame using ffmpeg, based on the aspect ratio dictated by the relevant parameter in the user configuration. Overall, the pipeline undertakes either *ball-centered cropping*, where cropping dimensions are computed dynamically based on the coordinates created by the interpolation module (optimal operation), or *frame-centered cropping*, where cropping dimensions are computed statically based on the aspect ratio and frame dimensions, to point to the mathematical frame center (fallback in case of too few ball detections, or explicit user configuration *not* to use "smart" crop). As **output**, SmartCrop returns an .mp4 file generated from the cropped frames. It also has additional functionality to prepare processed data for visualization, summarization, and further analysis.

3 Evaluation

To evaluate SmartCrop, we performed both objective and subjective experiments. Figure 3 presents the **object detection** performance of various YOLO configurations. Notably, `Sc11` significantly outperforms all other models. In this experiment, we monitored the precision and recall across training epochs to ensure effective learning without overfitting. Starting from epoch 1 with a precision of 0.71 and recall of 0.046, there was a gradual increase, achieving a precision of approximately 0.969 and a recall rate of 0.896 by epoch 109. This trend suggests that the model is effectively learning to identify more positive samples (high precision) and capturing a larger proportion of total positive samples (high recall). To guard against overfitting, we implemented several strategies, including regular validation on a separate dataset, early stopping if the validation loss ceased to decrease, and a dropout rate.

To evaluate the **outlier detection** and **interpolation** performance, we assessed the Root Mean Square Error (RMSE) [4] of various methods against a meticulously annotated ground truth derived from a 30-second soccer video, detailing the ball's position in each frame. Following the validation, the IQR method demonstrated superior accuracy and has since been integrated as the default approach in our pipeline. Among alternative interpolation methods, heuristic interpolation [6] proved to perform best and has been adopted as the default.

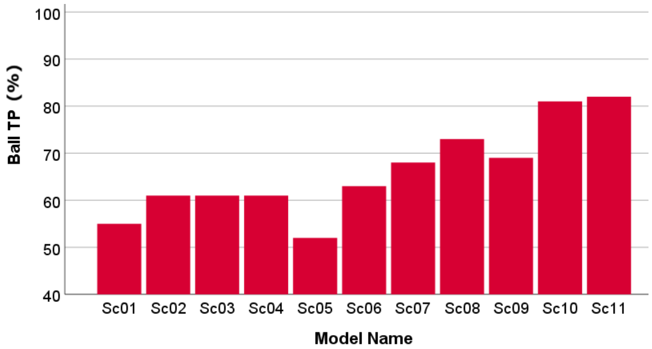


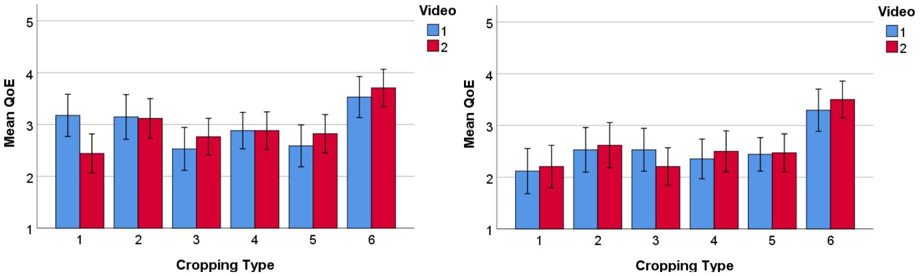
Fig. 3: The performance of various YOLO models in terms of ball true positive (TP) detections. Note that the y-axis starts at 40% to better highlight differences.

To evaluate **system performance**, we tested a local deployment on an NVIDIA GeForce GTX 1050 GPU with 4GB memory, focusing on the impact of the Skip Frame parameter. Most modules, such as scene detection and cropping, maintained consistent execution times across different configurations. However, by adjusting the Skip Frame parameter from 1 to 13, the object detection module's execution time decreased 70% due to processing fewer frames.

To assess the **end-to-end subjective performance** of the pipeline, we performed a user survey via Google Forms to retrieve feedback from participants in a crowd-sourced fashion on the final cropped video output. We compared 2 static and 4 dynamic approaches, as shown in Table 1. Two representative videos were selected to evaluate each approach, one of normal gameplay with fast motion and edge field features (video 1) and one of a goal with varying motion and ball occlusion features (video 2), each cropped to 1:1 and 9:16 target aspect ratios. Thus, four cases were constructed overall (two videos, each cropped to two aspect ratios) to compare the performance of six different cropping approaches (where the full SmartCrop corresponds to approach 6). We recruited 23 participants (9 female, 14 male) with ages ranging from 18 to 63 (mean: 30.95), all of whom were active on social media. For each case, the participants first viewed the original video, and then each of the processed videos corresponding to cropping approaches 1-6, which they rated using a 5-point Absolute Category Rating (ACR) scale, as recommended by ITU-T P.800. As shown in Figure 4, one-way repeated measures ANOVA results indicate an overall improved QoE for both videos in both aspect ratios, for SmartCrop (approach 6).

No	Crop Centering	Description	Outlier detection	Interpolation
1	frame-centered	static no padding	✖	✖
2	frame-centered	static w/black padding to 16:9	✖	✖
3	ball-centered	use last detected ball position	✖	✖
4	ball-centered	w/interpolation	✖	✔
5	ball-centered	w/outlier detection	✔	✖
6	ball-centered	w/interpolation & outlier detection	✔	✔

Table 1: Cropping alternatives used in the subjective evaluation.



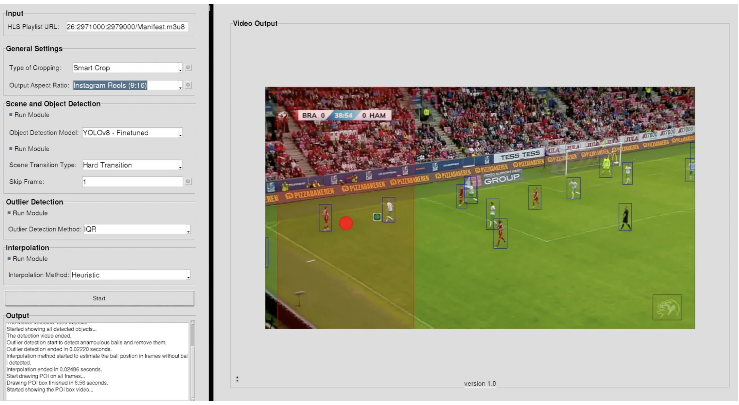
(a) Target aspect ratio 1:1

(b) Target aspect ratio 9:16

Fig. 4: QoE ratings for different target aspect ratios with 95% confidence intervals.

4 Demonstration

We demonstrate the SmartCrop pipeline with a graphical user interface (GUI) as depicted in Figure 5. First, the participants set up the cropping operation by selecting an HLS stream and setting configuration parameters such as output aspect ratio, skip frame, object detection model, outlier detection model, and interpolation model. Then, the pipeline can be started, which runs in three steps; i) scene and object detection; ii) outlier detection and interpolation; and iii) cropping and video output. For each of the steps, our GUI provides debugging information (e.g., execution time in the text output box), as well as intermediate results in terms of visual output (video output window), to demonstrate the individual contributions of various modules. Finally, the cropped video can be viewed. A video of the demo can be found here: https://youtu.be/aqqPWfrPmsE, with more details on the pipeline in [6].



Input

HLS Playlist URL: 26:2971000:2979000/Manifest.m3u8

General Settings

Type of Cropping: Smart Crop

Output Aspect Ratio: Instagram Reels (9:16)

Scene and Object Detection

■ Run Module

Object Detection Model: YOLOv8 - Finetuned

■ Run Module

Scene Transition Type: Hard Transition

Skip Frame: 1

Outlier Detection

■ Run Module

Outlier Detection Method: IQR

Interpolation

■ Run Module

Interpolation Method: Heuristic

Output

```
Started showing all detected objects...
The detection video ended.
Outlier detection start to detect anamoulous balls and remove them.
Outlier detection ended in 0.02220 seconds.
Interpolation method started to estimate the ball postion in frames without bal
l detected.
Interpolation ended in 0.02486 seconds.
Start drawing POI on all frames...
Drawing POI box finished in 6.56 seconds.
Started showing the POI box video...
```

Fig. 5: SmartCrop GUI - shown here is an intermediate result with POI markings.

References

1. Apostolidis, K., Mezaris, V.: A fast smart-cropping method and dataset for video retargeting. In: Proc. of IEEE ICIP. pp. 2618–2622 (2021)
2. Bourke, P.: Interpolation methods. Miscellaneous: projection, modelling, rendering **1**(10) (1999)
3. Castellano, B.: SceneDetect. https://github.com/Breakthrough/ PySceneDetect/tree/main (2023)
4. Chai, T., Draxler, R.R.: Root mean square error (RMSE) or mean absolute error (MAE)? Arguments against avoiding RMSE in the literature. Geoscientific model development **7**(3), 1247–1250 (2014)
5. Deselaers, T., Dreuw, P., Ney, H.: Pan zoom scan – time-coherent trained automatic video cropping. In: Proc. of IEEE CVPR. pp. 1–8 (2008)
6. Dorcheh, S.M.M., Sarkhoosh, M.H., Midoglu, C., Sabet, S.S., Kupka, T., Riegler, M.A., Johansen, D., Halvorsen, P.: SmartCrop: AI-based cropping of soccer videos. In: Proc. of IEEE ISM (2023)
7. Gautam, S.: Bridging multimedia modalities: enhanced multimodal AI understanding and intelligent agents. In: Proc. of ACM ICMI (2023)
8. Gautam, S., Midoglu, C., Shafiee Sabet, S., Kshatri, D.B., Halvorsen, P.: Assisting soccer game summarization via audio intensity analysis of game highlights. In: Proc. of 12th IOE Graduate Conference. vol. 12, pp. 25 – 32. Institute of Engineering, Tribhuvan University, Nepal (2022)
9. Gautam, S., Midoglu, C., Shafiee Sabet, S., Kshatri, D.B., Halvorsen, P.: Soccer game summarization using audio commentary, metadata, and captions. In: Proc. of ACM MM NarSUM. pp. 13–22 (2022)
10. Husa, A., Midoglu, C., Hammou, M., Halvorsen, P., Riegler, M.A.: HOST-ATS: Automatic thumbnail selection with dashboard-controlled ML pipeline and dynamic user survey. In: Proc. of ACM MMSys. p. 334–340 (2022)
11. Husa, A., Midoglu, C., Hammou, M., Hicks, S.A., Johansen, D., Kupka, T., Riegler, M.A., Halvorsen, P.: Automatic thumbnail selection for soccer videos using machine learning. In: Proc. of ACM MMSys. p. 73–85 (2022)
12. Jain, E., Sheikh, Y., Shamir, A., Hodgins, J.: Gaze-driven video re-editing. ACM TOG **34**(2), 1–12 (2015)
13. Jocher, G., Chaurasia, A., Qiu, J.: Ultralytics YOLOv8. https://github.com/ultralytics/ultralytics (2023)
14. Kaur, H., Kour, S., Sen, D.: Video retargeting through spatio-temporal seam carving using kalman filter. IET Image Processing **13**(11), 1862–1871 (2019)
15. Kemper, M., Rosso, G., Monnone, B., Kemper, M., Rosso, G.: Creating animated effects. Advanced flash interface design pp. 255–288 (2006)
16. Kopf, S., Haenselmann, T., Kiess, J., Guthier, B., Effelsberg, W.: Algorithms for video retargeting. Multimedia Tools Appl **51**(2), 819–861 (2011)
17. Lee, H.S., Bae, G., Cho, S.I., Kim, Y.H., Kang, S.: Smartgrid: Video retargeting with spatiotemporal grid optimization. IEEE Access **7**, 127564–127579 (2019)
18. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. In: Proc. of IEEE ICCV. pp. 2980–2988 (2017)
19. Liu, F., Gleicher, M.: Video retargeting: automating pan and scan. In: Proc. of ACM MM. pp. 241–250 (2006)
20. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., Berg, A.C.: SSD: Single shot multibox detector. In: Proc. of ECCV. pp. 21–37 (2016)

21. Midoglu, C., Sabet, S.S., , Sarkhoosh, M.H., Dorcheh, S.M.M., Gautam, S., Kupka, T., Halvorsen, P.: AI-based sports highlight generation for social media. In: Proc. of ACM MHV (2024)
22. Nam, H., Park, D., Jeon, K.: Jitter-robust video re-targeting with kalman filter and attention saliency fusion network. In: Proc. of IEEE ICIP. pp. 858–862 (2020)
23. Nergård Rongved, O.A., Hicks, S.A., Thambawita, V., Stensland, H.K., Zouganeli, E., Johansen, D., Midoglu, C., Riegler, M.A., Halvorsen, P.: Using 3D convolutional neural networks (CNN) for real-time detection of soccer events. International Journal of Semantic Computing **15**(2), 161–187 (2021)
24. Nergård Rongved, O.A., Hicks, S.A., Thambawita, V., Stensland, H.K., Zouganeli, E., Johansen, D., Riegler, M.A., Halvorsen, P.: Real-time detection of events in soccer videos using 3D convolutional neural networks. In: Proc. of IEEE ISM. pp. 135–144 (2020)
25. Nergård Rongved, O.A., Stige, M., Hicks, S.A., Thambawita, V.L., Midoglu, C., Zouganeli, E., Johansen, D., Riegler, M.A., Halvorsen, P.: Automated event detection and classification in soccer: The potential of using multiple modalities. Machine Learning and Knowledge Extraction **3**(4), 1030–1054 (2021)
26. Noorkhokhar: YOLOv8-football: How to detect football players and ball in real-time using YOLOv8: A computer tutorial. https://github.com/noorkhokhar99/YOLOv8-football
27. Rachavarapu, K.K., Kumar, M., Gandhi, V., Subramanian, R.: Watch to edit: Video retargeting using gaze. Computer Graphics Forum **37**, 205–215 (2018)
28. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once (YOLO): Unified, real-time object detection. In: Proc. of IEEE CVPR. pp. 779–788 (2016)
29. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: Towards real-time object detection with region proposal networks. Advances in neural information processing systems **28** (2015)
30. Roboflow: Football players detection dataset. https://universe.roboflow.com/roboflow-jvuqo/football-players-detection-3zvbc (2023)
31. Saleem, S., Aslam, M., Shaukat, M.R.: A review and empirical comparison of universe outlier detection methods. Pakistan journal of statistics **37**(4) (2021)
32. Sarkhoosh, M.H., Dorcheh, S.M.M., Gautam, S., Midoglu, C., Sabet, S.S., Halvorsen, P.: Soccer on social media. arXiv preprint arXiv:2310.12328 (2023)
33. Soucek, T., Lokoc, J.: TransNet V2: an effective deep network architecture for fast shot transition detection. CoRR (2020)
34. Valand, J.O., Kadragic, H., Hicks, S.A., Thambawita, V., Midoglu, C., Kupka, T., Johansen, D., Riegler, M.A., Halvorsen, P.: Automated clipping of soccer events using machine learning. In: Proc. of IEEE ISM. pp. 210–214 (2021)
35. Valand, J.O., Kadragic, H., Hicks, S.A., Thambawita, V.L., Midoglu, C., Kupka, T., Johansen, D., Riegler, M.A., Halvorsen, P.: AI-based video clipping of soccer events. Machine Learning and Knowledge Extraction **3**(4), 990–1008 (2021)
36. Wang, C.Y., Bochkovskiy, A., Liao, H.Y.M.: YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. arXiv preprint arXiv:2207.02696 (2022)
37. Wang, S., Tang, Z., Dong, W., Yao, J.: Multi-operator video retargeting method based on improved seam carving. In: Proc. of IEEE ITOEC. pp. 1609–1614 (2020)
38. Wang, Y.S., Lin, H.C., Sorkine, O., Lee, T.Y.: Motion-based video retargeting with optimized crop and warp. In: Proc. of ACM SIGGRAPH. pp. 1–9 (2010)