Optimal Crop-Out for Photographing People during Sporting Activities

Anastasia Lebedenko*
Supervised by: prof. Ing. Adam Herout Ph.D.†

Faculty of Information Technology
Brno University of Technology
Brno / Czech Republic

Abstract

This paper presents a solution for processing footage featuring human subjects to generate videos of optimal dimensions, focused on the individual, and eliminating redundant background. Utilizing computer vision models, the program identifies and tracks human positions in the input videos, then applies a specialized cropping algorithm to generate output frames. The solution offers customization options for aspect ratio, crop mode, and graphic overlay in the output video. Thus, it eliminates the necessity for capturing multiple videos to meet varied technical or aesthetic requirements, allowing the creation of diverse outputs from a single high-resolution video using predefined cropping parameters.

Keywords: computer vision, cropping algorithm, video processing

1 Introduction

Capturing videos of individuals in motion is challenging due to potential issues of exiting the frame or being disproportionately small compared to their environment [11]. The objective is to simplify the filming process by allowing users to capture one extensive video, and subsequently process it to meet diverse specifications, including adjusted frame size, aspect ratio, or focusing on specific segments of the human body. Such functionality enables generation of multiple customized video outputs from a single source file.

The solution requires developing an algorithm for precise Region of Interest (ROI) identification within each frame and a cropping strategy that ensures consistent positioning of the ROI across frames. The quality output video should appear stable from frame to frame, without visible jumps, that can be induced by frame cropping [12].

Existing video cropping solutions lack automation and comprehensive coverage of the human body (as discussed in Section 2).

The developed automated program, discussed in this paper, offers multiple cropping parameters and modes, ensuring the output video is stable and visually appealing. The solution eliminates the need for specialized recording equipment.

2 Existing Solutions

Incorporating the essential feature of cropping entire video clips, a number of video editing platforms, such as Final Cut Pro, extend the functionality to manually modify cropping parameters for individually segmented portions of video [6].

Adobe Premiere Pro employs an automatic AI-powered Auto Reframe feature [5], which crops footage to fit specified aspect ratios. This tool leverages motion tracking to accurately identify and maintain the visibility of the ROI throughout changes in frame resolution, ensuring critical elements remain within view in the output video. The process is predominantly automated, users are given the option to fine-tune the result by selecting among three predefined levels of camera motion intensity.

Apple's Center Stage feature [7] is a solution for real-time video crop. Available on select devices with an ultra-wide camera, it dynamically centers people on the camera preview, e.g. during video calls.

A state-of-the-art solution is Cloudinary API [2], that offers a large variety of crop modes as well as AI technology to gravitate video crop to the pre-determined ROI: faces or other user-specified objects. Despite its capabilities, this solution does not prioritize achieving an optimal frame size, which is the key feature of the proposed program. Moreover, it does not guarantee consistent inclusion of the entire subject within the frame or accommodate specific body capture orientations, such as portrait mode.

Reliance on AI for video processing, as highlighted in Adobe's documentation [5], may introduce artifacts upon recurrent processing of identical footage. Moreover, there currently exists no commercial or open-source program that replicates the unique approach of combining machine learning with direct mathematical cropping. The proposed

*xlebed11@vutbr.cz
†herout@fit.vut.cz

program offers a high degree of customization of processing parameters without the risk of significant artifacts.

3 Proposed Optimal Crop Algorithm

The proposed video processing algorithm, as shown in Figure 1, utilizes a two-phase architecture. In the scope of the initial video processing, it extracts body landmarks to generate bounding and frame box coordinates, and stores these 3 types of coordinates in separate JSON files with uniform structure, shown in Figure 1. This step, crucial due to its resource-intensive nature, ensures that landmark detection is only conducted once per video, thereby optimizing the cropping process for repeated crops of the same footage.

3.1 Detection

The program uses two MediaPipe detection solutions: Pose Landmark detection [9] and Object detection [8].

The program uses two MediaPipe detection solutions: Pose Landmark Detection and Object Detection. BlazePose, the underlying technology for Pose Landmark Detection, employs a lightweight convolutional neural network (CNN) architecture [1]. It combines heatmaps and regression to keypoint coordinates, enabling the detection of up to 33 body landmarks and the generation of a segmentation mask for a single person.

During inference, BlazePose adopts a detector-tracker setup. Initially, a body pose detector identifies the person in the frame. This is followed by a pose tracker network which predicts keypoint coordinates and refines the region of interest for accurate pose tracking. The detector focuses on detecting a relatively rigid body part, like the torso, using a fast on-device face detector as a proxy. This innovative method overcomes the limitations of traditional Non-Maximum Suppression algorithms, which often struggle with the complexity of human poses. The pose estimation network then predicts the location of 33 keypoints based on the alignment provided by the detector, effectively capturing complex human movements with high precision.

While segmentation mask is redundant in terms of human detection, it aids to more precise result, comparing to other human detection solutions, that return bounding boxes with excessive space on the edges [4, 10]. The landmarks are useful for cropping video based on the body capture orientations. Moreover, the chosen API succeeds in differentiating the most prominent person on the frame, which is useful for videos, where individual sport is performed with audience in the background. On the other hand, this mechanism is not fit for partner sports, as the set of landmarks will be calculated for just one person.

For partner sports, e.g. dancing, the MediaPipe Object Detection API is used. Detector output includes a name of object category e.g "human" and dimensions of the detected bounding box. The API also proved useful in the experiments with cropping a video of a person together with sporting equipment (e.g. cycling - the output video was cropped based on the combined position of the person and bicycle)

3.2 Bounding Box Calculation

In one-person mode, the bounding box is a rectangle that encloses the contour of the segmentation mask, returned by the detector, as shown in Figure 2. In case only a part of body is needed for the crop, the lower boundary of the segmentation mask bounding rectangle is cropped based on the y-coordinate of the relevant body landmark (Figure 3).

In case of partner sports, the final bounding box is obtained by summing up the bounding boxes of the relevant classes ("human" is default, classes with the names of sporting equipment are optional).

3.3 Frame Box Calculation

As the human moves, the dimensions of the corresponding bounding box can change from frame to frame. To ensure all video frames in the cropped output maintain constant dimensions, calculating the frame boxes is crucial. The exact size of the frame box depends on the chosen crop mode (Section 3.4). In case of a fixed frame, the output video size is defined by the coordinates, that enclose the area where a human was present at some point throughout the whole video. Otherwise, the largest width and height value among all bounding boxes define the size of an output video, with regard to if a specific aspect ratio was chosen. As per Figure 4, the frame box coordinates are calculated such that the bounding box is centered inside it. Figure 5 showcases the example of such positioning on a sample video frame.

Aspect Ratio

Adjusting to specific aspect ratios during cropping can indeed present a complex challenge, necessitating a variety of methods as noted by existing research [3]. Nonetheless, presented program simplifies this process significantly. It allows for selective inclusion of the surrounding environment by altering the dimensions of the frame box around the bounding box, thus eliminating the need for the complex methodologies typically required. This approach provides flexibility in determining the extent and specific areas to be included around the subject, facilitating a more intuitive and efficient cropping process.

Edge Cases

For frames like in Figure 6a, when person is moving towards the edge of the frame, bounding box centering results in frame box values exceeding the dimensions of in-

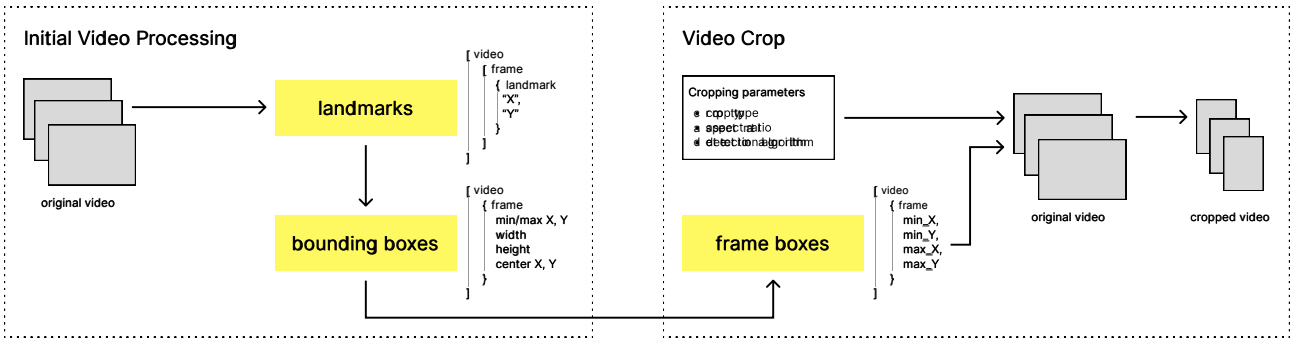

Figure 1: Architecture of the proposed solution. The first phase (**Initial Video Processing**) is executed once per unique video, and saves the output of processing (landmarks, bounding and frames boxes' coordinates). The second phase (**Video Crop**) utilizes the processed data to crop the original video based on the user-defined cropping parameters.

Figure 2: Segmentation mask with overlaid rectangle, which coordinates were calculated based on the edges of mask's contour. As a result, an optimal bounding box around the entire human body is found.

Figure 3: Bounding box, derived from the segmentation mask, cropped based on the y-coordinate of hips pose landmark.

put video. In this case, centering constraint is not included in the calculations.

For frames to be cropped and extracted from input video, each frame has to have frame box values defined. As shown in Figure 6b, if frame $n+1$ is missing frame box coordinates, these values are iteratively propagated from frame n and vice versa.

Stabilization

To ensure stable output video, crop-out values are filtered using Savitzky-Golay filter from SciPy library [13]. An array of each frame box coordinate's values in each frame is processed by savgol_filter( ) function.

3.4 Crop Modes

Crop modes are special crop settings implemented in the program, appropriate for specific type of movements in the input footage. Yoga videos, where person remains on the same place, could be cropped by **fixed frame**, which signifies the border, inside which all action is happening. For movements, that are primarily up and down, or left to right, **one-direction** cropping mode eliminates frame fluctuations in the secondary direction. Default **two-direction** mode can be combined with **zoom** option: in footage, where person distances from the camera, such mode zooms the frame in and out, so that person appears to be in the same distance.

4 Desktop Program

The solution is a Python-based desktop application that operates via a command-line interface, facilitating the cropping of video files through various parameters: input video(s) path, cropping mode, aspect ratio, body capture orientation (ranging from head-shot to full body), and optional graphical overlays (including detected landmarks, bounding, and frame boxes). Capable of batch processing entire directories, the program is designed for efficient handling of extensive video datasets.

5 Conclusion and Future Work

The solution effectively executes video cropping tasks across a wide array of video types, including scenarios featuring single or multiple subjects, with or without background activity. The qualitative evaluation of the desktop program is still in progress, with a primary focus on gathering user feedback regarding the appropriateness of different crop modes for different sports.

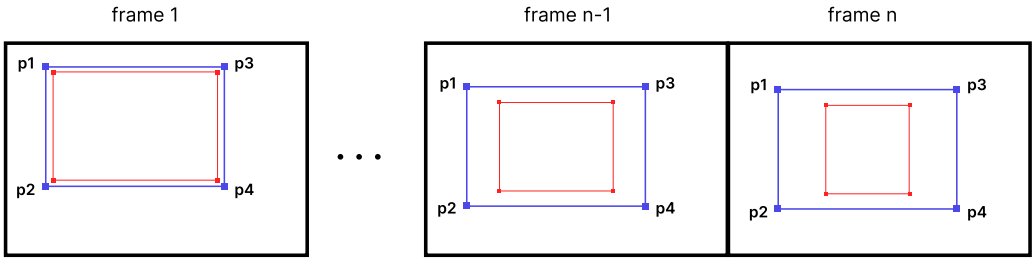

Figure 4: The dimensions of the frame box are determined by the largest values of width and height observed across all bounding boxes. For every frame that is cropped, the frame box is strategically positioned to ensure the bounding box remains centered within it.

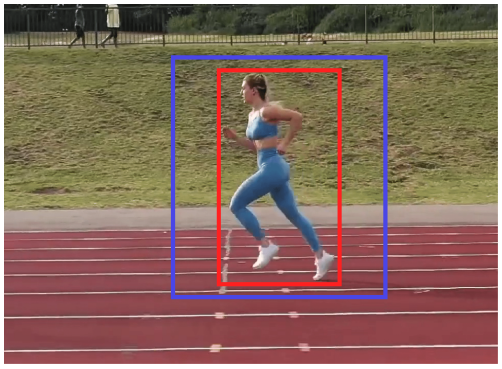

Figure 5: Bounding box and frame box on a sample frame.

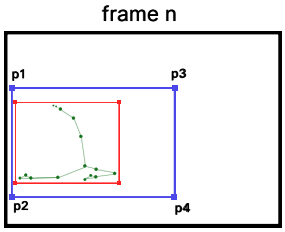

Figure 6a: Bounding box on the edge of input video, removal of centering constraint.

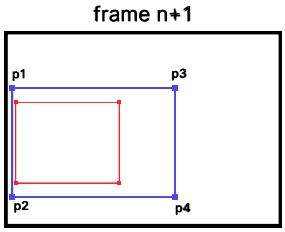

Figure 6b: Landmarks not detected, duplicate of previous frame box used.

While the program excels in its primary function of video cropping tailored to body dimensions, it lacks the comprehensive features of a full-fledged video editor, positioning it as a single-purpose tool ideal for batch processing, particularly in research contexts. It accepts user inputs through command-line arguments without providing a graphical user interface (GUI).

For convenient crop of videos, captured on smartphone, a logical improvement of the solution is development of a mobile application. Efforts will concentrate on integrating video cropping functionalities as well as devising a user-friendly interface that addresses the challenges of displaying complex cropping parameters on limited screen sizes, ensuring intuitive and visual editing workflows. Additional considerations include automatic video selection from device galleries, personalized cropping recommendations based on the video's characteristics or previous user settings, aiming to streamline the video cropping experience for end-users.

References

[1] Valentin Bazarevsky, Ivan Grishchenko, Karthik Raveendran, Tyler Zhu, Fan Zhang, and Matthias Grundmann. Blazepose: On-device real-time body pose tracking, 2020.

[2] Cloudinary. Video resizing and cropping. `https://cloudinary.com/documentation/video_resizing_and_cropping`. Accessed on 07.03.2024.

[3] Zhongliang Deng, Yandong Guo, Xiaodong Gu, Zhibo Chen, Quqing Chen, and Charles Wang. A comparative review of aspect ratio conversion methods. In *2008 International Conference on Multimedia and Ubiquitous Engineering (mue 2008)*, pages 114–117, 2008.

[4] Derek Hoiem, Yodsawalai Chodpathumwan, and Qieyun Dai. Diagnosing error in object detectors. In Andrew Fitzgibbon, Svetlana Lazebnik, Pietro Perona, Yoichi Sato, and Cordelia Schmid, editors, *Computer Vision – ECCV 2012*, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

[5] Adobe Inc. Automatically reframe video for social media channels. `https://helpx.adobe.com/premiere-pro/using/auto-reframe.html`. Accessed on 07.03.2024.

[6] Apple Inc. Crop clips in final cut pro for mac. `https://support.apple.com/cs-cz/guide/final-cut-pro/verb8e5db98/mac`. Accessed on 07.03.2024.

[7] Apple Inc. Use center stage on your ipad or studio display. `https://support.apple.com/en-us/HT212315`. Accessed on 07.03.2024.

[8] MediaPipe. Object detection task guide. https://developers.google.com/mediapipe/solutions/vision/object_detector/. Accessed on 07.03.2024.

[9] MediaPipe. Pose landmark detection guide. https://google.github.io/mediapipe/solutions/pose.html. Accessed on 07.03.2024.

[10] Duc Thanh Nguyen, Wanqing Li, and Philip O. Ogunbona. Human detection from images and videos: A survey. *Pattern Recognition*, 51:148–175, 2016.

[11] Manoj Ramanathan, Wei-Yun Yau, and Eam Khwang Teoh. Human action recognition with video data: Research and evaluation challenges. *IEEE Transactions on Human-Machine Systems*, 44(5):650–663, 2014.

[12] Yufei Xu, Jing Zhang, and Dacheng Tao. Out-of-boundary view synthesis towards full-frame video stabilization. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4822–4831, 2021.

[13] Çağatay Candan and Hakan Inan. A unified framework for derivation and implementation of savitzky–golay filters. *Signal Processing*, 104:203–211, 2014.