



# Automatic highlight detection in videos of martial arts tricking

Marcos Rodrigo<sup>1</sup> · Carlos Cuevas<sup>1</sup> · Daniel Berjón<sup>1</sup> · Narciso García<sup>1</sup>

Received: 18 March 2022 / Revised: 3 March 2023 / Accepted: 29 May 2023 /  
Published online: 21 July 2023  
© The Author(s) 2023, corrected publication 2023

## Abstract

We propose a novel strategy for the automatic detection of highlight events in user-generated tricking videos, to the best of our knowledge, the first one specifically tailored for this complex sport. Most current methods for related sports leverage high-level semantics such as predefined camera angles or common editing practices, or rely on depth cameras to achieve automatic detection. However, our approach only relies on the contents (themselves) in the frames of a given video, and consists in a four stage pipeline. The first stage identifies foreground key points of interest along with an estimation of their motion in the video frames. In the second stage, these points are grouped into regions of interest based on their proximity and motion. Their behavior over time is evaluated in the third stage to generate an attention map indicating the regions participating in the most relevant events. The fourth and final stage provides the extracted video sequences where highlights have been identified. Experimental results attest to the effectiveness of our approach, which shows high recall and precision values at frame level, with detections that fit well the ground truth events.

**Keywords** Highlight event detection · Automatic video summary · Temporal segmentation · Martial arts tricking

## 1 Introduction

The volume of video data generated has experienced exponential growth over the years [36]. This huge amount of data requires efficient mechanisms, such as automatic video summarization, to create efficient video representations to make it easier to browse, find, and manage digital media content [27]. Automatic video summarization is a process well known to be very time consuming [14]. Furthermore, it usually requires highly professional tools and

---

Carlos Cuevas, Daniel Berjón and Narciso García contributed equally to this work.

---

✉ Marcos Rodrigo  
marcos.rodrido@upm.es

Extended author information available on the last page of the article

expert editing skills [9]. Consequently, it is an important and growing research area where many different techniques have emerged over the past few years [5].

There are many applications for automatic video summarization, ranging from surveillance footage to broadcast events. In this work, we focus on its application specifically for the purpose of summarizing user-generated tricking videos. This is a relatively new sport that emerged around 2000 from martial arts exhibitions, when practitioners started incorporating different flips in their routines [17]. Martial arts tricking is mostly known as just “tricking”, and it incorporates moves from a mix of several different areas, including taekwondo, gymnastics and break dance among others. Players usually perform passes, also referred to as combinations or combos, in which they combine kicks, transitions, flips and twists in quick succession (see Fig. 1).

Tricking emerged on social media platforms, and the community primarily uses these platforms to share videos. In this community samplers are known as highlight reels where people summarize their progression over an interval of time. Given the nature of the sport, it is common for tricking practitioners to record their training sessions and competitions for future highlight reels or for learning purposes. This often results in lengthy video sequences in which passes represent only a small portion. Here is where automatic highlight detection algorithms come in handy.

The specific scenario that has been considered for this work has a static recording camera, which is the most common recording setup. For a scenario such as this, the camera is usually placed at a certain distance aiming at the center of the training environment and players take turns to perform their passes while the rest wait their turn in the background while moving or stretching. Players’ passes stand out from other events in terms of motion [12] as they execute skills that encourage fast movement and body extension. That is why the focus of this work is on estimating motion features from the players. Using this information, the proposed strategy can automatically extract highlights from a given video.

As will be further discussed on Section 2, there already is literature on video summarization and highlight detection in the gymnastics and martial arts areas [34, 40]. However, tricking



**Fig. 1** Example images of the sport of tricking. Players perform combinations of skills that mainly include kicks, flips, twists and transitions

brings new challenges that previous methods do not address, as movement is much more erratic and does not follow the strict patterns other related sports do. For instance, gymnasts always perform passes in a straight line while flipping and twisting only in the vertical axis, making it easier to detect patterns [2, 20]. Something similar happens with most martial arts, where practitioners try to achieve or hold specific poses that make pattern recognition an easier task [22, 35]. However, tricking players perform flips and twists in the vertical, horizontal, diagonal and even off axis, not always following a straight line, and poses are not nearly as strict as in most other martial arts, encouraging innovation and self-expression. These arguments show the big difference in complexity there exists between these sports when dealing with automatic highlight detection.

This paper presents the first strategy for automatic highlight detection in martial arts tricking, proposing a solution that is not based on deep learning and thus does not require massive datasets. This approach relies solely on the content of the frames, without the need for depth cameras or other equipment. In addition, it not only allows identifying events but modeling their relevance. It should also be noted that our work operates at the frame level, providing more detailed analysis than at the video segment level. The proposed strategy has been assessed on three video sequences with a total duration of 53 minutes, obtaining very high quality results. Overall, our proposed method offers a new and effective solution for tricking video analysis and has the potential to impact other sports as well.

The paper is organized as follows: Section 2 explores related works. Section 3 presents an overview of the strategy we propose, while sections 4, 5, 6 and 7 go into further detail about each step in the pipeline. Experimental results are presented in Sections 8, and 9 concludes the paper.

## 2 Related work

Due to the exponential growth in the volume of video data generated in recent years, several approaches have been developed to automatically generate summaries and detect highlights [18, 36]. Many of these approaches focus on the generation of video summaries of sports events, since their highlights often only account for a small portion of the total video length (e.g., in soccer [8] and cricket [37]).

Throughout this section the most prominent video summarization strategies for sports events are described, paying special attention to those most closely related to martial arts tricking.

Many of the proposed strategies are oriented to the generation of summaries of broadcast sports, which are typically based on the detection of the high-level semantics resulting from the video editing, such as the shot sizes, the on-screen graphics, or the wipe transitions [8]. In [11], a strategy to summarize broadcast soccer games is proposed, which is based on detecting specific elements (e.g., the goal posts) through the utilization of predefined camera angles in edited videos. The strategy in [26] allows detecting highlights in sports broadcasts by identifying slow-motion replays. In [41], a high-accuracy framework is proposed to automatically clip sports video streams using a three-level prediction algorithm based on YOLO-v3 and OpenPose.

It is also possible to find in the literature a large number of works oriented to the generation of summaries of user-generated sports videos, which are characterized by the lack of any standardized editing conventions or universal structure that can be leveraged to extract high-level semantics. Some of these works focus on reducing the redundancy in long videos

based on the analysis of low-level features, such as color [21] or significant objects [25]. In [34], an approach to summarize kendo videos is proposed, which detects players' actions using a deep neural network-based method that extracts two types of action-related features to classify video segments as highlights: body joint-based features and holistic features. In [40], a cross-category video highlight detection strategy is proposed, which addresses the problem of highlight detection in sports where the annotated videos required to train the algorithms are not available. The strategy in [3] enables highlight detection using a network that leverages a bimodal attention mechanism to capture the relationship between the audio and visual components. In [4], a simple contrastive learning framework is proposed for unsupervised video highlight detection that encodes a video into a vector representation by learning to pick video clips that help distinguish it from other videos. The strategy in [39] uses an encoder-decoder network with 3D convolutional neural networks (CNNs) and visual saliency that learns pixel-level distinctions to improve the detection of interesting segments. The strategy in [23] proposes a method that addresses the challenges of cross-modal representation learning and fine-grained feature discrimination. This method uses intra-modality encoding and cross-modality co-occurrence encoding to augment features and capture effective information, and a hard-pairs guided contrastive learning scheme to improve feature discrimination.

Although most video summarization strategies focus on the sports with the highest popularity (e.g., soccer, basketball, baseball, and tennis) [30], there is also a significant amount of algorithms that focus on other sports. In [13], an inter-frame similarity algorithm that compares each frame with keyframes given by a human is used to summarize gymnastics and figure skating videos that contain multiple shots, camera motion, and dynamic movements of objects. A strategy to automatically identify trampoline skills with a single camera using CNN-based pose estimators is proposed in [7]. In [38], an end-to-end CNN capable of operating a camera motion control system is used to record or broadcast rhythmic gymnastics highlights. In [19], it is proposed a strategy focused on the analysis of taekwondo. This strategy relies on a Structure Preserving Object Tracker (SPOT) that allows the target player to be tracked and segmented, feeding frames that fully contain the player's body to a deep learning network (PCANet) that predicts the player's actions. At a later stage, it uses a linear support vector machine (SVM) to classify techniques into groups of frames rather than individual frames. The strategy in [28] evaluates the performance of a gymnast on the pommel horse apparatus utilizing a depth camera. This approach identifies a depth of interest in the RGB-D frame, localizes the gymnast, detects when the gymnast is performing a certain routine, and provides an analysis of that routine. In [10] a method is proposed for the automatic recognition and scoring of Olympic rhythmic gymnastics. This method extracts detailed velocity field information from body movements and transforms them into spatiotemporal image templates that are automatically assigned a score by comparing them to a set of stored movements of the same type that have been assigned scores by expert judges. The work in [42] introduces a new dataset called AGF-Olympics for athlete performance measurement in sports videos. This dataset incorporates artistic gymnastic floor routines and provides highly challenging scenarios with extensive background, viewpoint, and scale variations. To analyze the videos, this work proposes a discriminative attention module to map the dense feature space into a sparse representation by disentangling complex associations.

In conclusion, prior work in sports video summarization has largely focused on well-established sports such as soccer, basketball, and tennis, relying on common editing practices or predefined camera angles. However, these methods do not work well for tricking, a sport that presents unique challenges such as rapid movements, constant changes of direction, and spontaneous and unstructured movement sequences that do not adhere to predetermined

patterns. Deep learning approaches have shown promising results in video summarization, but these methods are heavily reliant on large and diverse datasets, which are not readily available for smaller sports such as tricking. Likewise, approaches based on depth cameras are not well-suited for tricking, as accurately tracking the body joints becomes challenging due to the fast and intricate movements involved. Therefore, novel and customized techniques are required to handle the challenges specific to tricking (and other smaller sports), such as the one proposed in this paper.

### 3 System overview

Highlight events in the sport of martial arts tricking consist in players performing passes. Passes can have different duration and incorporate a variety of skills, but they all share that during a pass, a player combines different skills in quick succession. So it is the fast motion of a player performing which makes a highlight event stand out from other events. We propose a strategy able to extract such information to automatically identify highlight events from a given video.

The basic outline of our strategy consists in four processing blocks, as shown in Fig. 2, each with a well-defined task.

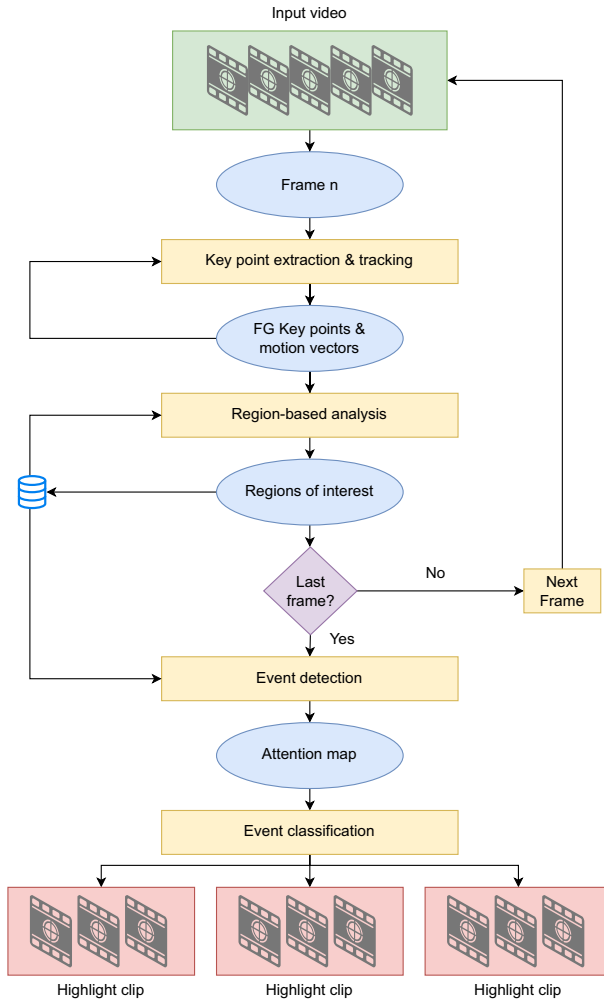
In the key point extraction and tracking block (Section 4), the most prominent corners, or key points, of a frame are extracted and filtered from well-known background key points. This filtering is based on a probabilistic model dependent on the location history of the extracted set of key points, to focus on the region of interest (i.e., the foreground where players are performing). The resulting foreground key points are tracked from frame to frame to estimate their motion vectors, and both foreground key points along with their motion vectors will serve as low-level features that capture players' motions.

In the region-based analysis block (Section 5), foreground key points along with their motion vectors are grouped up into regions, which account for the spatial relationship amongst key points of the same frame, and summarize the information provided by the sparse set of key points that form them in a more compact way.

The event detection block (Section 6) comes after all frames of the input video have been analyzed for regions, and it is in charge of identifying the events they participate in. These will serve to generate an attention map that indicates, for each frame, the region participating in the most relevant event, under the assumption that the region of a frame participating in the most relevant event suffices to determine whether or not the entire frame can be classified as a highlight or not later on.

Finally, the event classification block (Section 7) analyzes the information contained in the regions spanned by the attention map to perform an initial binary classification at the frame level, classifying frames as either highlight or not. This initial classification is followed by a refinement stage in which highlight frames close in time are grouped forming highlight events, for which we model their relevance to produce the final result. The final result consists in a set of video sequences extracted from the input video where highlight events have been identified.

Following our proposed method, we are able to establish a correspondence between the initial key points (first system block) and the final events identified. Initial key points exclusively capture spatial information, while additional higher-level semantics are incorporated through the processing performed in the following blocks, including essential temporal information, which is key for the identification of highlights.



**Fig. 2** Block diagram of the proposed strategy. Rectangular blocks denote processing blocks, round-edge blocks denote data, and diamond blocks denote decision making

## 4 Key point extraction and tracking

This processing block extracts a set of key points from a frame (subsection 4.1), separating the key points belonging to the foreground of the scene from those that are part of the background (subsection 4.2), and tracks the foreground key points from frame to frame (subsection 4.3) to estimate their motion vectors.

### 4.1 Key point extraction

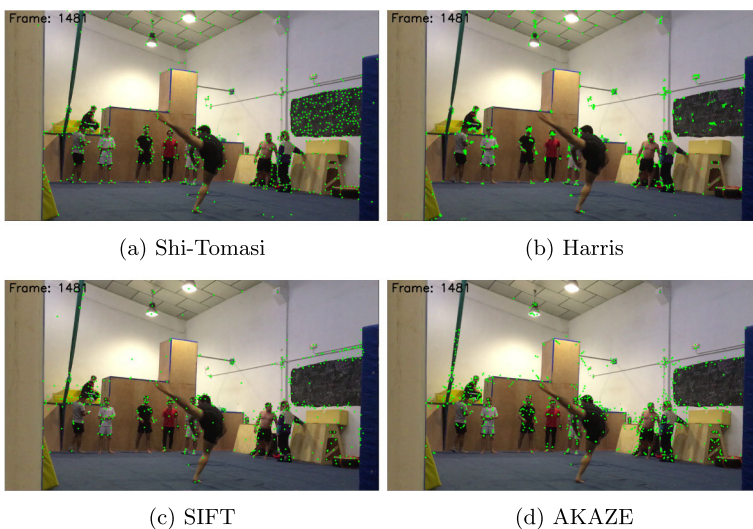
Key points are the lowest-level features of the proposed strategy and they serve to identify locations of interest in an image. For instance, let  $I^n$  be the current image being analyzed

at time  $n$ , and let  $(x^n, y^n)$  be a pixel with column  $x^n$  and row  $y^n$ . Locations of interest are identified as a set of  $K^n$  key points,  $Q^n = \{(x_k^n, y_k^n), 1 \leq k \leq K^n\}$ , where the pair  $(x_k^n, y_k^n)$  represents the coordinates (column and row) of the  $k$ -th key point extracted from the current image,  $I^n$ . The strategy we propose builds on top of these key points to extrapolate higher-level semantics.

Many relevant key point extraction methods were evaluated to determine the most suitable one for our strategy. Such method should be able to extract enough and well distributed key points from players performing, and these should be stable for tracking purposes. Moreover, this work contemplates video sequences where in addition to the player performing a highlight pass, there are other people moving in the foreground and the background, and so, the best fitting method should also be able to extract enough and well distributed key points over the entire frame. This allows to characterize both the players (i.e., foreground) and the environment (i.e., background).

Among others, we performed extensive experiments with Shi-Tomasi [31], Harris [15], SIFT [24], and AKAZE [1] algorithms. Figure 3 illustrates the key points obtained with these methods after tuning them for the most favorable results possible. Shi-Tomasi provides a significant amount of key points well distributed throughout the frame and the players, and these are stable for tracking. Harris provides key points that are well distributed over the frame (although appearing in dense clusters, adding for some redundancy) but not over the players. Additionally, these are not stable for tracking, as they cannot be found on players performing due to their blurriness, which makes corners appear more diffuse. SIFT provides a stable set of key points, but these are ill-distributed over the frame and the players (e.g., ceiling and players' feet show very few key points). Finally, AKAZE provides a stable set of key points well-distributed over the players, but not over the frame, and similarly to SIFT, it also struggles extracting key points from players' feet.

Thus, the Shi-Tomasi algorithm proved to be the most adequate solution for extracting the set of key points  $Q^n$  for our strategy, as it identifies a sufficient quantity of key points that are well-distributed over the frame and the players. Additionally, these key points are stable across



**Fig. 3** Key points detected using different feature extraction methods

video frames, which facilitates smooth tracking. The algorithm detects the most prominent corners in a gray-scale representation of the frame by identifying little image patches or windows that generate significant variations in intensity when moved around.

## 4.2 Key point filtering

The set of key points  $Q^n$  is filtered from well-known background ( $BG$ ) key points,  $Q^n_{BG}$ , to focus on the region of interest (i.e., the players performing). This filtering reduces the data to process and also prevents matching foreground ( $FG$ ) key points with well-known  $BG$  key points when tracking them in the next processing step, described in subsection 4.3.

Similar to Sun et al. [32], we use a method to update the probability of each pixel of being part of the  $BG$ , based on the location history of the extracted set of key points,  $Q^n$ . The probability of a pixel  $(x^n, y^n)$  at frame  $I^n$  being part of the  $BG$  is computed as

$$\Pr^n(x^n, y^n) = \begin{cases} \Pr^{n-1}(x^n, y^n)\lambda + (1 - \lambda), & (x^n, y^n) \in Q^n \\ \Pr^{n-1}(x^n, y^n)\lambda, & (x^n, y^n) \notin Q^n \end{cases} \quad (1)$$

where  $\lambda$  is a learning factor set to 0.95 as suggested in [29]. Following Eq. (1), if a key point is consistently identified in the same location across frames, the probability of such pixel of being part of the  $BG$  will increase. Therefore, a set of filtered key points is obtained as

$$Q^n_f = \{(x^n, y^n) \in Q^n \mid \Pr^n(x^n, y^n) < T\} \quad (2)$$

where  $T$  is a threshold value (set at the empirical value of 0.15) that filters well-known  $BG$  key points. As depicted in Fig. 4,  $BG$  key points (in red) can be found on static objects of the scene as well as on players who stay immobile for a period of time. On the other hand, the remaining set of key points (in green) can be found on moving objects and, due to illumination changes, on some static objects. However, as described in subsection 4.3, the latter will be easily removed.



**Fig. 4** Example of well-known  $BG$  key points filtering. Red represents well-known  $BG$  key points while green represents the remaining set of key points



### 4.3 Tracking

The set of filtered key points,  $Q_f^n$ , is matched against the set  $Q_f^{n-1}$ , corresponding to the previous frame, to estimate the motion vectors associated to each key point in  $Q_f^n$ , by making use of the iterative Lucas-Kanade method with pyramids [6].

Key points presenting very small motion vector magnitudes (e.g., less than 2 pixels) are filtered to mitigate illumination changes previously mentioned in subsection 4.2. This filtering prevents static key points from being interpreted as moving due to changes in illumination, and results in the set of *FG* key points,  $Q_{FG}^n$ . Figure 5 shows an example of the results obtained following this method. The interesting region of the scene (i.e., the player performing) is captured by the set of *FG* key points along with their motion vectors. It can be seen that after filtering key points with little motion associated, all key points that were incorrectly classified as part of the *FG* in the previous stage have been discarded.

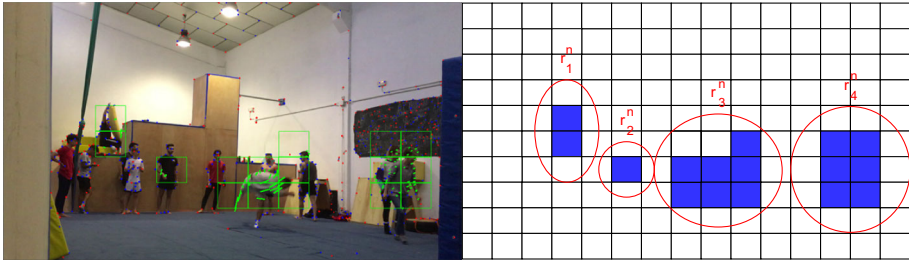
## 5 Region-based analysis

*FG* key points,  $Q_{FG}^n$ , along with their associated motion vectors, are grouped up by vicinity forming a set of regions. The underlying idea is that the motion of these regions can provide more robust and reliable information than that provided by the sparse key points that form them, allowing to identify and characterize interesting regions of the scene. It is worth noting that players are neither rigid nor always present the same orientation. Therefore, it is not feasible to establish durable key point correspondences throughout a sequence, which motivates the proposed region-based approach.

For this purpose  $I^n$  is tessellated into a grid of non-overlapped uniform cells  $C_{\{W,H\}}$  (e.g., a grid of  $15 \times 10$  cells), where  $W$  and  $H$  represent a cell location, column and row, respectively. Cells are sized large enough to represent a region but small enough to provide local information. The set of *FG* key points,  $Q_{FG}^n$ , is mapped onto these cells revealing which ones are active (i.e., cells containing at least 1 *FG* key point).



**Fig. 5** Example of motions estimated using a pyramidal implementation of the Lucas-Kanade algorithm. In green the set of motion vectors associated to *FG* key points, in blue those with very small magnitudes, and in red well-known *BG* key points



**Fig. 6** Example of identified regions using the proposed method. Foreground key points are mapped to their corresponding cells and four regions of different sizes are identified (enumerated from 1 to 4 from left to right)

Active cells in the grid are grouped up by vicinity following a connected-component labelling approach [16], resulting in a set of  $J^n$  regions,  $R^n = \{r_j^n, 1 \leq j \leq J^n\}$ , where  $r_j^n$  represents the  $j$ -th region of the  $n$ -th frame. Regions are characterized by the cells that form them ( $C_{\{W,H\}}$ ), by the number of  $FG$  key points that fall within them ( $N$ ) along with the sum of their associated motion vectors magnitudes ( $S$ ), and by their normalized motion ( $\bar{M} = S/N$ ). When taking motion into account we do not consider its direction but only its magnitude, as this suffices to represent the overall motion present in a region.

Intuitively, fast-moving objects are commonly blurrier than objects moving slower. Therefore, it is likely that less key points will be extracted from them in the first place (see subsection 4.1), and these are likely to present larger motions. The normalized motion of a region,  $\bar{M}$ , allows to compare the amplitude of the motion between regions irrespective of the number of detected key points.

In the example of Fig. 6 we can distinguish four regions that correspond, from left to right, to a person moving an object, the waving of a hand of another person, a person performing a pass, and two people walking together. The information each region contains is summarized in Table 1. Regarding  $N$  and  $S$ , regions 3 and 4 are much more relevant than the other two regions. This alone manifests the capability of this method to identify foreground regions of interest and characterize them. In addition,  $\bar{M}$  shows its usefulness for differentiating regions more relevant in terms of motion. Region 3 having half as many points as region 4, but an overall motion twice as large, presents a normalized motion almost four times greater.

### 5.1 Region filtering

Regions can be further filtered at this processing block in order to save some computational cost at later blocks, storing only regions of interest that are good candidates to be a part of a

**Table 1** Regions description.  $r_j^n$ :  $j$ -th region of the  $n$ -th frame.  $C_{\{W,H\}}$ : active cells.  $N$ : number of key points.  $S$ : sum of associated motion vectors magnitudes (pixels).  $\bar{M}$ : normalized motion (pixels/key point)

$r_j^n$	$C_{\{W,H\}}$	$N$	$S$	$\bar{M}$
$r_1^n$	$C_{\{3,4\}}, C_{\{3,5\}}$	4	22.55	5.64
$r_2^n$	$C_{\{5,6\}}$	3	11.70	3.90
$r_3^n$	$C_{\{7,6\}}, C_{\{7,7\}}, C_{\{8,6\}}, C_{\{8,7\}}, C_{\{9,5\}}, C_{\{9,6\}}, C_{\{9,7\}}$	26	273.11	10.50
$r_4^n$	$C_{\{12,5\}}, C_{\{12,6\}}, C_{\{12,7\}}, C_{\{13,5\}}, C_{\{13,6\}}, C_{\{13,7\}}$	46	130.40	2.83

highlight. This is done under two simple assumptions: (1) good candidate regions are at least of a certain size and (2) are likely to be located around the same area than a good candidate region of the previous frame.

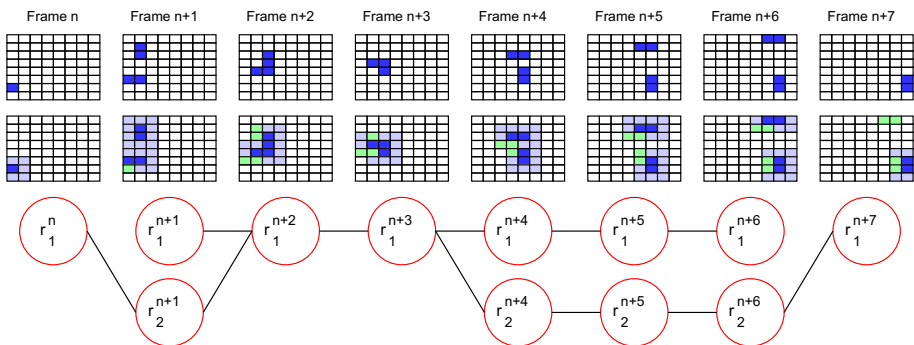
The first assumption is easy to interpret. A region of interest must contain at least a minimum number of cells, as these are dimensioned large enough so that they can represent a region but not so large that they can be of interest by themselves. Regions too small to feature a person (e.g., 2 cells or less) are discarded as they surely do not represent a good candidate region to be a part of a highlight, which usually entails long motion vectors that involve several different cells.

The second assumption is based on the underlying idea that regions of interest will smoothly evolve along frames, so it is highly unlikely for a region to appear at a location in a frame and in the next one be at a completely different location. This allows removing regions that appear sporadically over frames by seeking neighboring cells with regions of interest identified in the previous frame and discarding those which do not share neighbors.

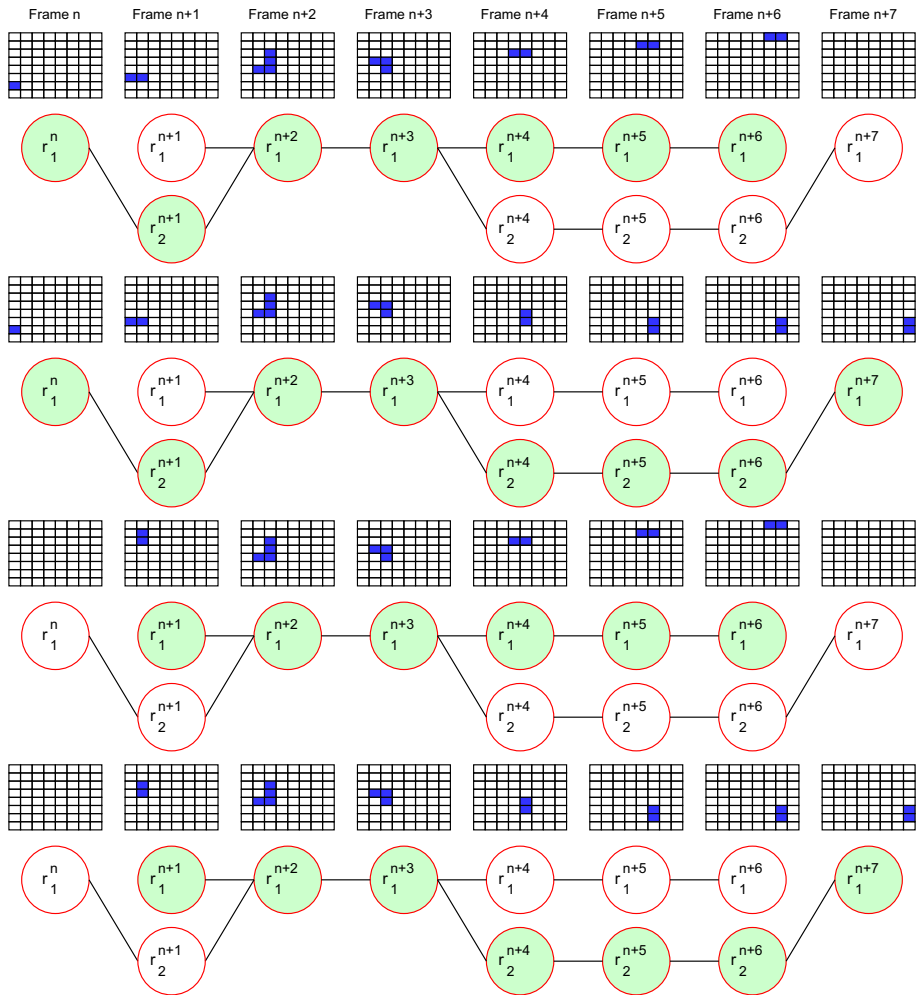
### 6 Event detection

This processing block is executed after all frames of the input video have been analyzed for regions (see Section 5), and it is in charge of identifying the events they participate in. These will serve to generate an attention map that indicates, for each frame, the region participating in the most relevant event. This is done under the assumption that the region of a frame participating in the most relevant event suffices to determine whether or not the entire frame can be considered as part of a highlight (see Section 7).

The region-based analysis described in Section 5 summarizes the foreground motion information of each frame in a set of regions uniquely identified. To reveal possible events, regions of contiguous frames which share neighboring cells are first linked as depicted in Fig. 7 to analyze their evolution over frames: how they appear, disappear, displace, split, or merge.



**Fig. 7** Example of a set of linked regions. In blue are represented the active cells that form each region. Purple displays the neighborhood around each region. Green overlays the regions’ locations of the previous frame. The bottom graph represents the regions that are linked



**Fig. 8** All possible events identified among the linked regions of the original sequence of Fig. 7 following a DAG approach. Four events are identified, enumerated from  $E_1$  to  $E_4$  from top to bottom. Bottom figures illustrate the paths followed from start to end nodes. Top figures illustrate the regions associated to each of those paths

All these possible events can be revealed following a directed acyclic graph (DAG<sup>1</sup>) approach. Each node represents a region that is directed to other nodes (regions) of the following frame. Start nodes are those regions that do not have any links with the previous frame, whereas end nodes are those that do not have them with the following one.

Taking into account all start and end nodes we can define all series of unique events  $E_i$  as depicted in Fig. 8. Events consist in a set of linked regions which represent how a region evolves through frames from its appearance to its disappearance, and are characterized by the motion features contained in the regions they span. Different events can overlap in frames

<sup>1</sup> A graph consisting of nodes that are directed from one to another such that following those directions will never form a closed loop.

and share one or several regions, but there cannot be two identical events with the same set of linked regions. Additionally, very short events (e.g., less than 1 second) are removed at this processing block, similarly to what we did in subsection 5.1, as they are not likely to represent a highlight event.

## 6.1 Attention map

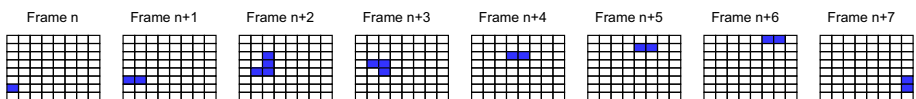
Under previous assumptions, the best candidate region of a frame to be a part of a highlight event would be that with the largest normalized motion. But temporal information plays a key role when assessing motion and has to be accounted for too. For this reason, we generate an attention map that indicates, for each frame, the region participating in the event that averages the largest normalized motion along the regions it spans. This allows selecting those regions participating in the most significant events, even when these regions do not show the largest normalized motion for a particular frame. This will serve as a cue to detect the start and end of a highlight event as will be further explained in Section 7.

The output of this processing block can be formulated as an attention map that indicates, at each frame, where the region more likely of being a part of a highlight event is, as depicted in Fig. 9. Comparing this figure with previous Fig. 8, it can be appreciated that when multiple events concur at a frame, the selected region is that which participates in the event that averages the largest normalized motion. For instance, let the average normalized motion of  $E_1$  be the largest of the four events. For frame  $n + 1$ , where the four events concur, region  $r_2^{n+1}$  is selected over  $r_1^{n+1}$  as it participates in the most relevant event. In frame  $n + 7$  only two events concur,  $E_2$  and  $E_4$ , and both share the same region, so  $r_1^{n+7}$  is selected for that frame.

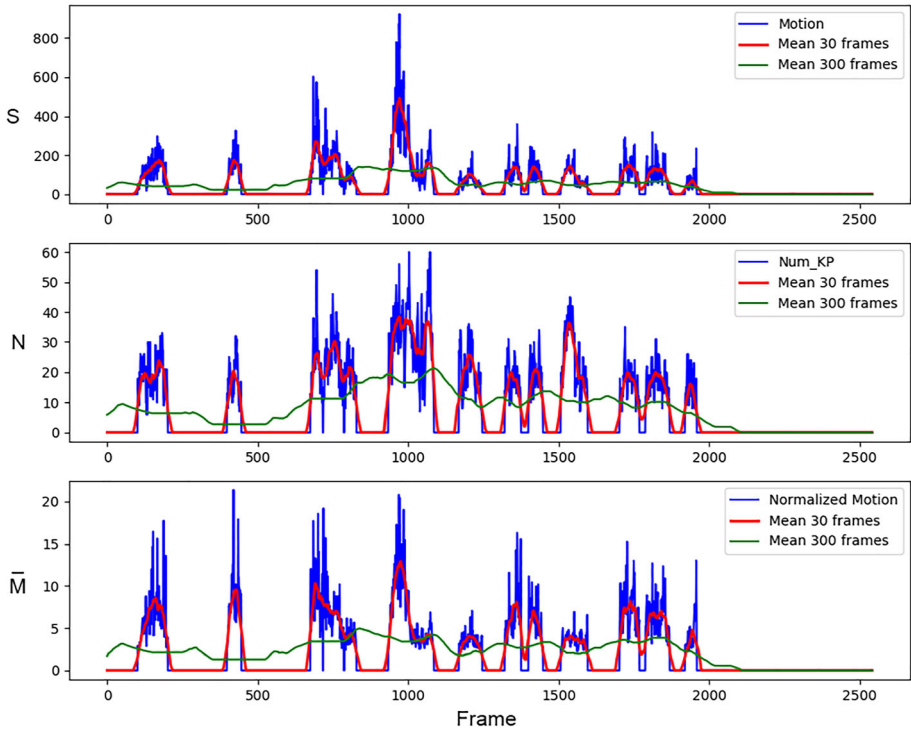
## 7 Event classification

Event classification constitutes the last processing block of the proposed strategy. It performs after each frame of the input video has been assigned a single region (or no region if none was identified) that participates in the most relevant event, as indicated by the attention map obtained in subsection 6.1. The motion information these regions contain is used to perform an initial binary classification at the frame level, classifying frames as either highlight or not. Highlight frames that are close in time are grouped together to form highlight events during a subsequent refinement stage, for which we model their relevance to produce the final result. The final result consists in a set of video sequences extracted from the input video where highlight events have been identified.

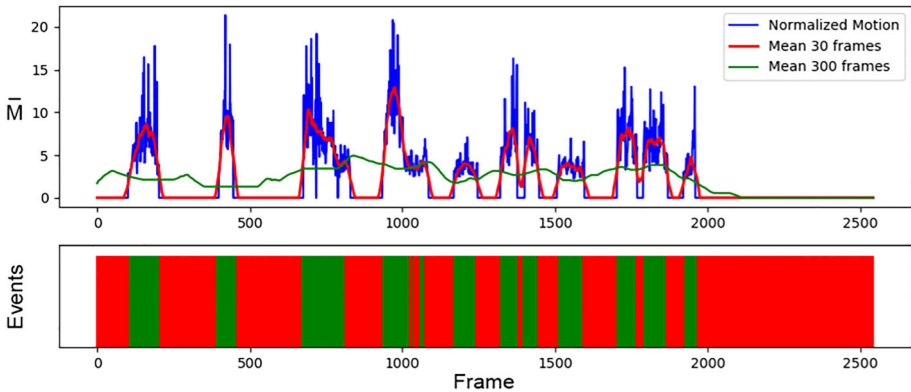
Figure 10 illustrates the motion information contained in the regions indicated by the attention map, where the three curves represent the values of  $S$ ,  $N$ , and  $\bar{M}$  over frames. In blue are represented the instantaneous values at each frame, which correspond to those of the region indicated by the attention map for that frame. It can be appreciated that these



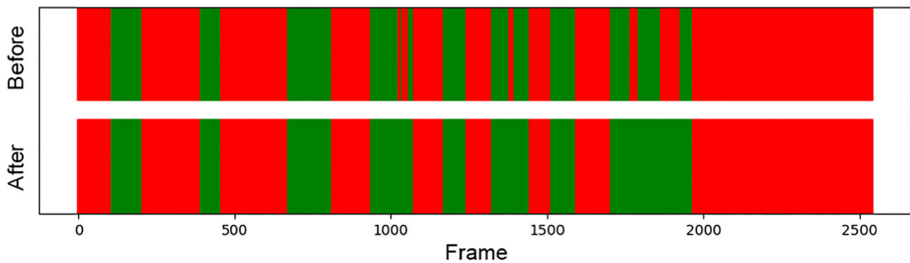
**Fig. 9** Example of generated attention map, which indicates, for each frame, the region participating in the event that averages the largest normalized motion



**Fig. 10** Summary of the motion information stored in the regions of the attention map. The three graphics correspond to the variables  $S$ ,  $N$  and  $M$ , respectively. The current values at each frame, which correspond to those of the region selected for that frame, are represented in blue. In red and green their rolling averages for a short and medium time windows, respectively



**Fig. 11** Initial binary classification of frames. Frames where the short-term average normalized motion exceeds the long-term (i.e., red line above green line) are initially classified as highlights, represented as green bars. Red bars correspond to frames classified as not a highlight



**Fig. 12** Grouping of highlight frames that are close in time, and thus, are likely part of the same highlight event. Top bar represents the classified frames before the closing operation, whereas the bottom bar represents the results after the operation, the highlight events

values remain zero for many frames, for which no regions of interest have been identified on previous blocks. In red and green are represented the rolling averages of these values for short and medium time windows (1 and 10 seconds) respectively, which will serve to measure how much a short event centered at a particular frame stands out from its surroundings.

### 7.1 Initial classification

An initial binary classification is performed at the frame level, classifying frames where the short-term average normalized motion exceeds the long-term (i.e., red line above green line) as highlight, or as non-highlight otherwise (see Fig. 11). Thus, frames are classified as highlights if a short event (1 second) centered around them stands out from its surroundings (10 seconds) in terms of normalized motion.

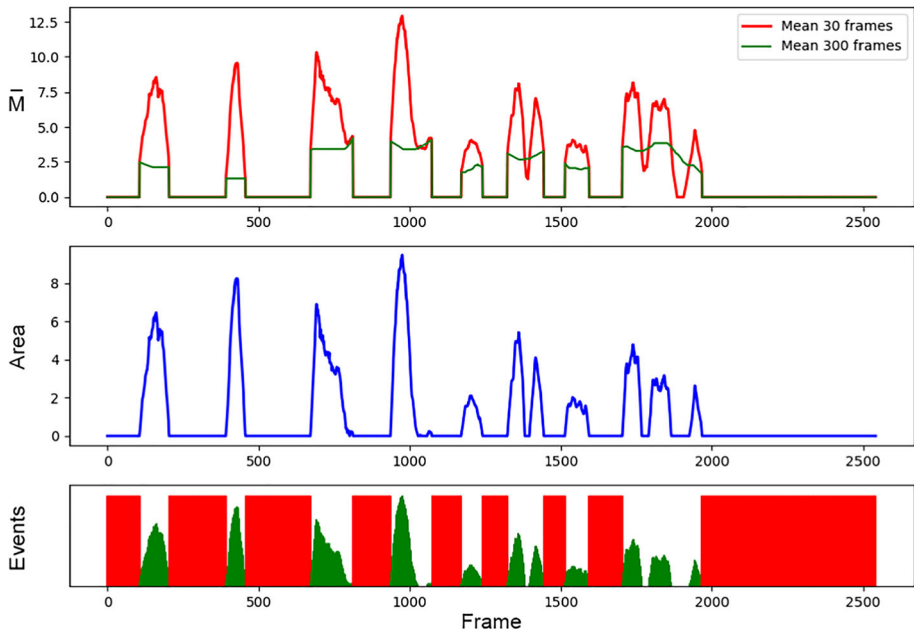
### 7.2 Classification refinement

The initial classification is first refined by extracting the set of identified highlight events from the grouping of highlight frames that are close in time (e.g., less than 1 second apart) and therefore are likely to belong to the same highlight event, as shown in Fig. 12.

The relevance of identified highlight events is modeled using the area enclosed between the short-term average normalized motion and the long-term, as depicted in Fig. 13. This area serves as a measure of how much identified highlight events stand out from their surrounding, making it possible to characterize their relevance. The relevance modelling of the identified highlight events constitutes an addition to their previous detection, making it possible to compare, order, or filter identified highlight events based on their relevance.

The final refinement step consists in the filtering of identified highlight events that are poor candidates, either because of their short duration (e.g., less than a second<sup>2</sup>) or because of their low relevance. To better illustrate this process, Fig. 14 shows the highlight events obtained after filtering poor candidates, along with the ground truth annotations, which will be covered in subsection 8.1. The final result produced by the proposed strategy is a set of

<sup>2</sup> Note that even though events of less than a second were filtered in Section 6, highlight events of shorter duration can appear in the initial frame classification performed in subsection 7.1.



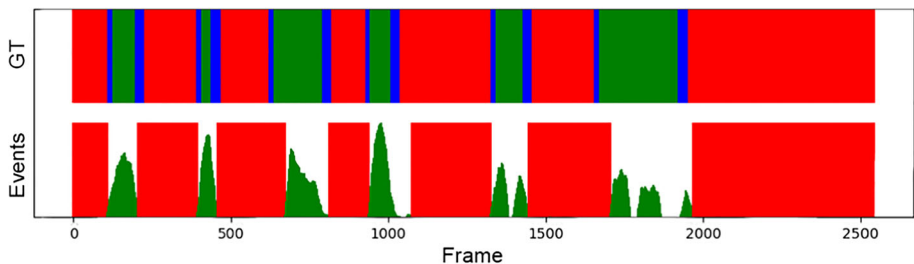
**Fig. 13** **a** Short- and long-term average normalized motions for a short and a medium time windows where highlight events have been identified. **b** Enclosed area. **c** Events after modelling their probability of being a highlight

video sequences comprised of frames from the input video where highlight events have been identified.

## 8 Results

### 8.1 Dataset

No publicly available dataset regarding tricking exists as of today and so, along this work a dataset was created from scratch. This dataset consists in three video sequences that represent



**Fig. 14** Final result of the proposed strategy after filtering poor candidate highlight events. The top bar represents the ground truth events, whereas the bottom bar corresponds to the identified highlight events. Red indicates not a highlight, green indicates a highlight and blue indicates uncertainty



**Table 2** Description of the three videos chosen to best represent the most common scenarios found in tricking

Video	Duration (min.)	Num. people	Highlights	Background
V1	21	7	Short duration, slow paced	Mostly static
V2	22	6	Medium duration, fast paced	Mostly static
V3	10	12	Long duration, fast paced	Highly dynamic

the most common scenarios found in tricking. Additionally, ground truth events have been manually annotated frame by frame<sup>3</sup> by an expert in tricking. These annotations include 3 different categories: highlight (HL), non-highlight (NHL), and uncertainty (UN). Only highlight events of a minimum duration of one second have been annotated, since events of shorter duration can hardly be considered a highlight.

The existence of uncertainty as a ground truth event is due to the nature of the sport, which makes it challenging to determine the exact frames where a highlight event starts or ends (even for an expert). Determining the start of a pass is usually easier as the player starts in a static position before performing, whereas the end of a pass is more difficult to establish as the player is usually carried by his momentum even after finishing the pass. This is why the ground truth was manually labeled using 0.5 and 1 seconds of uncertainty before and after every highlight event. Uncertainty serves as a margin of error in the manual labeling of ground truth events and will not penalize the results obtained.

The main characteristics of the three videos in the dataset are summarized in Table 2. The first two videos show normal training sessions with limited participants, the main difference between them being that in the first scenario highlights are mainly composed of single skills performed slowly (a common scenario when practicing new or specific skills), while in the second scenario, highlights involve longer duration and more motion, which is a common scenario for practicing new or specific skills (e.g., preparing performances for future competitions). The third scenario corresponds to a small gathering, where the number of people present is greater and players perform their best combinations of skills in quick succession.

Table 3 summarizes the ground truth events of the three videos. It can be easily noticed that highlight events only constitute a small fraction of the videos, around 10-23%, which manifests the necessity of algorithms that can automatically identify highlight events. For the first two videos, which show less number of people and where players perform more relaxed, highlights correspond to only around 10-15% of the videos and the average duration of these events is around 3 seconds. For the third video, which shows a significant increase in the number of people and in movement, highlight events constitute 23% of the entire video and the average duration of these highlights is longer.

## 8.2 Experiments

The results obtained from the final classification described in Section 7.2 have been divided into frame-level and event-level results. Frame-level results consist in the recall, precision, and F-score obtained from the comparison at the frame level between the identified events and the ground truth. These results provide an indication of the overall performance of the proposed

<sup>3</sup> The dataset and its ground truth are available at [www.gti.ssr.upm.es/data](http://www.gti.ssr.upm.es/data).

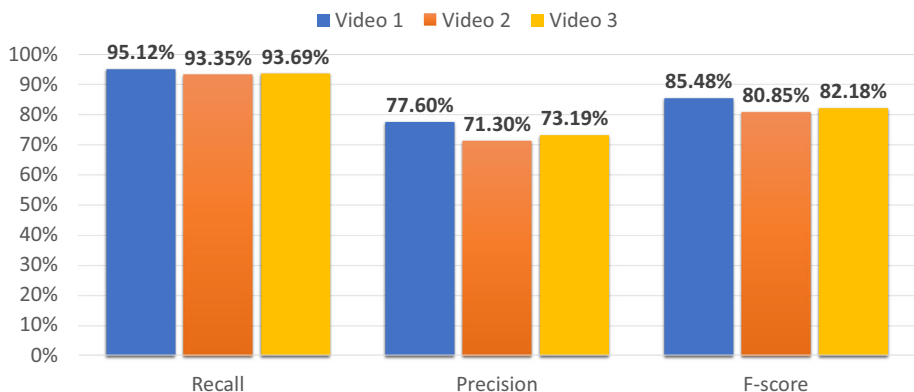
**Table 3** Ground truth events of the three chosen videos

Video	Event type	Avg. event duration (sec.)	Total event duration (min.)	% of the video
V1	HL	2.93	3.12	14.97 %
	NHL	14.88	16.13	77.35 %
	UN	0.75	1.60	7.68 %
V2	HL	2.69	2.29	10.43 %
	NHL	21.16	18.34	83.74 %
	UN	0.75	1.28	5.83 %
V3	HL	3.02	2.31	23.08 %
	NHL	8.55	6.56	65.43 %
	UN	0.75	1.15	11.49 %

strategy. However, they do not account for how identified events relate to their corresponding ground truth events (e.g., how many ground truth highlight events were correctly detected?). For that purpose, event-level results are provided.

Figure 15 summarizes frame-level results obtained for the three videos in the dataset. Their average values are: recall of 94.05%, precision of 74.03%, and F-score of 82.84%. Video 1 gives the best results (F-score of 85.48%), as it displays less movement and highlight events stand out more. Videos 2 and 3 present similar results (F-score above 80%), showing slightly less precision as a result of the increase in movement displayed in these videos, which makes more difficult to detect the start and end of events.

Event-level results obtained for the three videos in the dataset are summarized in Table 4. The total number of ground truth highlight events is 161. From the 171 highlight events identified, 158 are true positives (TPs), and 13 correspond to false positives (FPs). Additionally, there are only 3 false negatives (FNs). These results yield an average F-score of 95.18%. Out of the 13 FPs, 3 correspond to people crossing in front of the camera, 6 correspond to people crossing the scene at a fast pace after a long period of time without any highlight events, 3 correspond to players feinting a skill, and 1 corresponds to a celebration of a player's

**Fig. 15** Frame-level results consisting in recall, precision, and F-score obtained at frame level for the three different videos

**Table 4** Event-level results obtained for the three different videos

Video	GT highlights	Detected highlights	TP	FP	FN	F-score
V1	64	67	64	3	0	97.71 %
V2	51	54	50	4	1	95.24 %
V3	46	50	44	6	2	91.67 %
Total	161	171	158	13	3	95.18 %

performance by the other players. The 3 FNs correspond to 2 single-skill performances that were very short and occurred around much more relevant highlight events, and to 1 attempt to start a performance by a player that was interrupted to be executed again (the second time being correctly identified as a highlight event).

Additionally, the relevance modelling of highlight events described in Section 7.2 allows us to analyze the results obtained when sorting the identified highlight events by order of relevance<sup>4</sup>. Table 5 summarizes some of the event-level results obtained for different sets of the most relevant events identified. The top 100% corresponds to the results obtained when taking all identified events into consideration, and match those of Table 4. The lower percentages correspond to the results obtained for smaller sets of identified events, after discarding the less relevant ones. It can be appreciated that most FPs correspond to less relevant events, as for the 80% most relevant events identified only 2 correspond to FP events. The best F-score achieved is of 96.91% for the 95% most relevant events identified, which indicates that removing the 5% least relevant events identified could lead to better results, as this is where most erroneous detections occur.

The same results of Table 5 are illustrated in Fig. 16, where it is easier to appreciate that, as we expected, smaller sets of more relevant highlight events identified show higher precision but lower recall. Recall steadily increases as more identified highlight events are taken into consideration. However, precision slightly drops in the 84–100% range, as most FPs correspond to less relevant highlight events identified. As stated earlier, the best F-score is achieved for the 95% most relevant events, and past this point it drops as a consequence of the FPs introduced by the least relevant highlight events identified.

Figure 17 illustrates some images corresponding to highlight events that have been correctly detected (TPs), while Figs. 18 and 19 illustrate some images that correspond to FP and FN events, respectively.

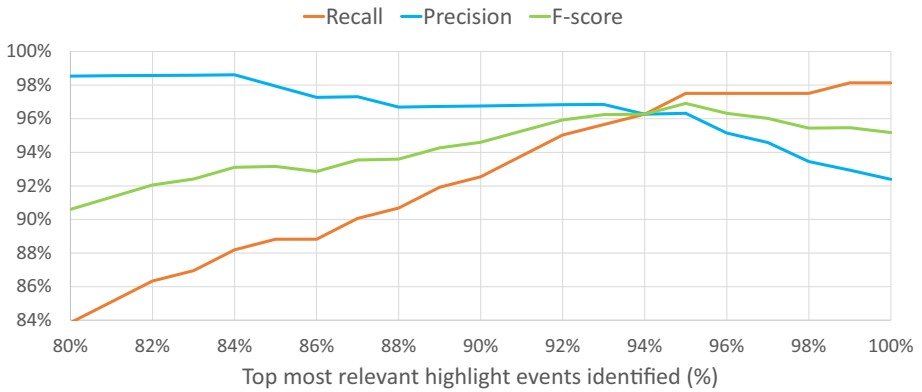
### 8.3 Comparison with other strategies

As stated in Section 2, no prior research has been conducted with the specific objective of detecting highlights in martial arts tricking. However, there are some strategies that focus on the detection of highlights in similar sports. Among these strategies, the recently proposed Dual-Learner-based Video Highlight Detection (DL-VHD) strategy [40] is the only one that can be applied to the detection of highlights in martial arts tricking, since it allows extracting highlights from a target video category by transferring the highlight knowledge acquired from a source category. This avoids the need to have a large amount of annotated videos of the same type as those to be analyzed.

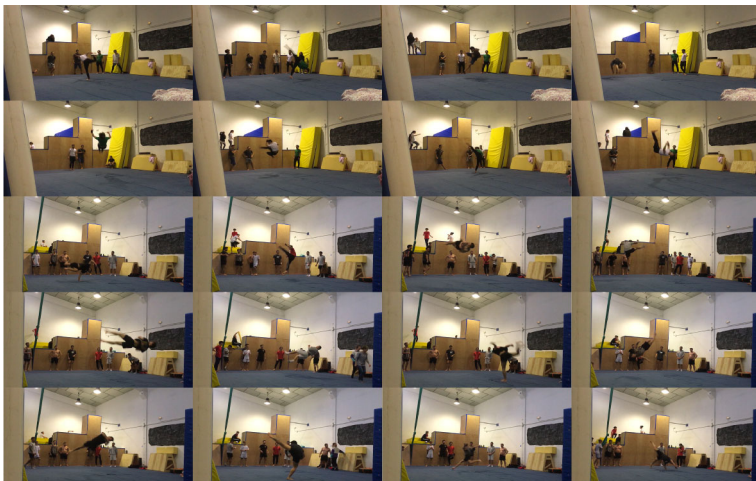
<sup>4</sup> Although ground truth highlight events are not characterized by a relevance value (as it would be highly subjective), we can still sort the identified highlight events by order of relevance to analyze if the events that were assigned a higher value are indeed more likely to correspond to a ground truth highlight.

**Table 5** Event-level results for different relevance ranges

Top % of the most relevant events identified	Detected highlights	TP	FP	FN	F-score
80 %	137	135	2	26	90.60 %
90 %	154	149	5	12	94.60 %
95 %	163	157	6	4	96.91 %
99 %	170	158	12	3	95.47 %
100 %	171	158	13	3	95.18 %



**Fig. 16** Event-level results obtained for different sets of the most relevant events identified



**Fig. 17** Example images corresponding to TP highlight events identified



**Fig. 18** Example images corresponding to FP highlight events identified

One of the primary discrepancies between our work and the DL-VHD strategy is in the annotation methodology. Whereas we provide frame-level annotations, the videos in the database used in DL-VHD have been annotated on a segment level [33]. Specifically, each video segment contains 100 frames and overlaps the previous segment by 50%. Using this overlap poses difficulties for the generation of a final video summary, since a significant number of frames can potentially belong to two video segments classified differently (i.e., one as HL and the other as NHL).

In an effort to compare our results with those obtained with the DL-VHD strategy, we have converted our frame-level annotations to the segment-level annotations (similar to those in [33]) required by the neural network architecture proposed in [40]: each segment has been assigned the label (HL, NHL, or UN) that appears most frequently within the 100 frames that constitute it.

In [40], the dataset used for the experiments is composed of different categories of videos manually annotated [33]. Among these categories, we have selected the two that share the most characteristics with martial arts tricking: gymnastics and parkour. Table 6 summarizes the mean average Precision (mAP), which is the metric used in the DL-VHD strategy, for different combinations of source and target categories. The upper part of the table shows that when the combined categories are gymnastics and parkour, the obtained mAP values are around 0.7. However, the results at the bottom of the table show that when the target category is tricking, the mAP values are much lower. This is mainly due to typical challenges



**Fig. 19** Example images corresponding to FN highlight events

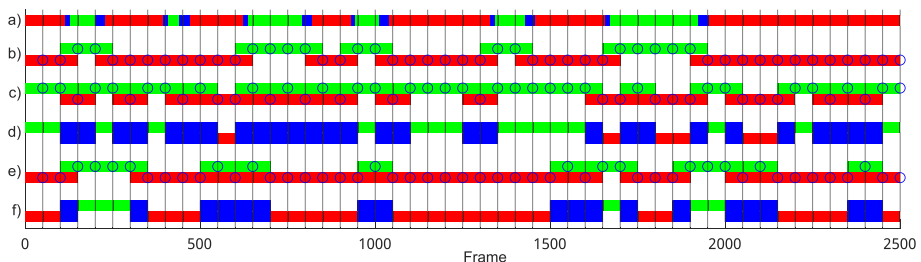
**Table 6** Results provided by the strategy in [40] for different source and target categories

Source category	Target Category	mAP
gymnastics	parkour	0.660
parkour	gymnastics	0.704
gymnastics	tricking	0.256
parkour	tricking	0.305

in martial arts tricking videos that are not present in other sports, and also due to the highlight detection based on video segments, which is very dependent on the length of such highlights. It should be noted that the results corresponding to the martial arts tricking videos have been obtained only on the video V1, since the videos V2 and V3 have been used for training the neural network (along with the corresponding gymnastics and parkour categories), and therefore it would not be fair to use them to extract results.

Figure 20 details the result of the classification provided by the DL-VHD strategy for the first 2500 frames of the sequence V1 (same range of frames illustrated in Fig. 14). These results have been obtained manually selecting the threshold that yields the highest F-Score. The top of this figure shows the original ground truth at the frame level (Fig. 20a) and the ground truth at the level of the partially overlapping segments (Fig. 20b). In Fig. 20c and e the results of the classification provided by the DL-VHD strategy have been represented when the gymnastics and parkour categories are used as a source, respectively. In addition, the result of converting these segment-level classifications to frame-level classifications is also illustrated (Fig. 20d and f). For this, all the frames with more than one label have been classified as UN (i.e., those classified as HL in one segment, but as NHL in another). This detailed representation shows that classification at the level of partially overlapping video segments is not suitable for detecting the highlights in this video.

Finally, Table 7 summarizes the recall, precision, and F-score frame-level values for sequence V1. These results show that the proposed strategy clearly outperforms the strategy in [40]. Furthermore, it is important to mention that unlike the strategy in [40], ours is capable of prioritizing the highlights detected by their relevance values.



**Fig. 20** Summary of the comparison process with [40]. **a** Original ground truth at frame level. **b** Ground truth adapted to a video segment level. **c** DL-VHD results when extrapolating highlights from gymnastics to tricking. **d** Same results as (c) but at the frame level. **e** DL-VHD results when extrapolating highlights from parkour to tricking. **f** Same results as (e) but at the frame level. Blue circles and vertical lines represent, respectively, the center and the limits of video segments. Green, red, and blue slices represent, respectively, HL, NHL, and UN events

**Table 7** Results obtained in V1 with the strategy in [40] and with the proposed strategy. The best results are highlighted in **bold**

Method	Source category	Target Category	Recall	Precision	F_score
DL-VHD	gymnastics	tricking	0.722	0.273	0.396
DL-VHD	parkour	tricking	0.319	0.519	0.398
Ours	—	tricking	<b>0.951</b>	<b>0.776</b>	<b>0.855</b>

## 9 Conclusions

This paper proposes a novel strategy for the automatic detection of highlight events in user-generated tricking videos, the first one tailored for this complex sport. This strategy is built around players' motion features and consists of a four-stage pipeline that automatically identifies foreground key points of interest, estimates their motion in the video frames, groups them in regions of interest, evaluates their behavior over time to generate an attention map indicating the regions participating in the most relevant events, and finally provides the extracted video sequences where highlights have been identified. The strategy we propose relies only on the content of the frames of an input video. It uses very low-level features to extrapolate high-level semantics and, unlike emerging deep learning approaches, offers great explicability, making it easier to adapt to new environments. Experimental results verify the effectiveness of our approach, which shows very high recall and precision at the frame level, with detections that fit well the ground truth highlight events.

**Acknowledgements** This work has been partially supported by project PID2020-115132RB (SARAOS) funded by MCIN/AEI/10.13039/501100011033 of the Spanish Government.

**Funding** Open Access funding provided thanks to the CRUE-CSIC agreement with Springer Nature.

**Data availability** The dataset and its ground truth are available at [www.gti.ssr.upm.es/data](http://www.gti.ssr.upm.es/data).

## Declarations

**Competing interests** The authors have no competing interests to declare that are relevant to the content of this article.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

- Alcantarilla PF, Solutions T (2011) Fast explicit diffusion for accelerated features in nonlinear scale spaces. *IEEE Trans Pattern Anal Mach Intell* 34(7):1281–1298
- Amri-Dardari A, Mkaouer B, Nassib SH, Amara S, Amri R, Salah FZB (2020) The effects of video modeling and simulation on teaching/learning basic vaulting jump on the vault table. *Sci Gymnast J* 12(3):325–344


3. Badamdorj T, Rochan M, Wang Y, Cheng L (2021) Joint visual and audio learning for video highlight detection. In: IEEE/CVF International Conference on Computer Vision. pp 8127–8137
4. Badamdorj T, Rochan M, Wang Y, Cheng L (2022) Contrastive learning for unsupervised video highlight detection. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp 14042–14052
5. Basavarajaiah M, Sharma P (2019) Survey of compressed domain summarization techniques. *ACM Comput Surv* 52(6):1–29
6. Bouguet J-Y (2001) Pyramidal implementation of the Affine Lucas Kanade feature tracker description of the algorithm. Intel Corporation 5(1–10):4
7. Connolly PW, Silvestre GC, Bleakley CJ (2017) Automated identification of trampoline skills using computer vision extracted pose estimation. Preprint at <http://arxiv.org/abs/1709.03399>
8. Cuevas C, Quilón D, García N (2020) Techniques and applications for soccer video analysis: a survey. *Multimed Tools Appl* 79(39):29685–29721
9. Dange B, Kshirsagar D, Khodke H, Gunjal S (2022) Automatic video summarization for cricket match highlights using convolutional neural network. In: IEEE International Conference on Smart Technologies and Systems for Next Generation Computing. pp 1–7
10. Díaz-Pereira MP, Gomez-Conde I, Escalona M, Olivieri DN (2014) Automatic recognition and scoring of olympic rhythmic gymnastic movements. *Hum Mov Sci* 34:63–80
11. Ekin A, Tekalp AM, Mehrotra R (2003) Automatic soccer video analysis and summarization. *IEEE Trans Image Process* 12(7):796–807
12. Grassie KP (2017) Kinematics of the lower extremities during fundamental martial arts tricking techniques. Honors Scholar Theses (522)
13. Han B, Hamm J, Sim J (2011) Personalized video summarization with human in the loop. In: IEEE Workshop on Applications of Computer Vision. pp 51–57
14. Haq HBU, Asif M, Ahmad MB (2020) Video summarization techniques: a review. *Int J Sci Technol Res* 9:146–153
15. Harris CG, Stephens M (1988) A combined corner and edge detector. In: *Alvey Vision Conference*, vol 15. pp 10–5244
16. He L, Ren X, Gao Q, Zhao X, Yao B, Chao Y (2017) The connected-component labeling problem: a review of state-of-the-art algorithms. *Pattern Recogn* 70:25–43
17. Hnitska T, Zavatska L, Holub O (2017) History of tricking foundation as an extreme sport and its distribution aspects in Ukraine. *Physical Education, Sport and Health Culture in Modern Society* (3(39)):29–33
18. Hussain T, Muhammad K, Ding W, Lloret J, Baik SW, de Albuquerque VHC (2021) A comprehensive survey of multi-view video summarization. *Pattern Recogn* 109:107567
19. Kong Y, Wei Z, Huang S (2018) Automatic analysis of complex athlete techniques in broadcast taekwondo video. *Multimed Tools Appl* 77(11):13643–13660
20. Lei Q, Zhang H, Du J (2021) Temporal attention learning for action quality assessment in sports video. *SIViP* 15:1575–1583
21. Lienhart RW (1999) Dynamic video summarization of home video. In: *Storage and Retrieval for Media Databases 2000*, vol 3972. pp 378–389
22. Liu M, Zhang J (2022) Gesture estimation for 3D martial arts based on neural network. *Displays* 72:102138
23. Li S, Zhang F, Yang K, Liu L, Liu S, Hou J, Yi S (2022) Probing visual-audio representation for video highlight detection via hard-pairs guided contrastive learning. Preprint at <http://arxiv.org/abs/2206.10157>
24. Lowe DG (2004) Distinctive image features from scale-invariant keypoints. *Int J Comput Vision* 60(2):91–110
25. Meng J, Wang H, Yuan J, Tan Y-P (2016) From keyframes to key objects: Video summarization by representative object proposal selection. In: IEEE Conference on Computer Vision and Pattern Recognition. pp 1039–1048
26. Pan H, Van Beek P, Sezan MI (2001) Detection of slow-motion replay segments in sports video for highlights generation. In: IEEE International Conference on Acoustics, Speech, and Signal Processing, vol 3. pp 1649–1652
27. Raval KR, Goyani MM (2022) A survey on event detection based video summarization for cricket. *Multimed Tools Appl* 81(20):29253–29281
28. Reily B, Zhang H, Hoff W (2017) Real-time gymnast detection and performance analysis with a portable 3D camera. *Comput Vis Image Underst* 159:154–163
29. Senior A (2002) Tracking people with probabilistic appearance models. In: *ECCV Workshop on Performance Evaluation of Tracking and Surveillance Systems*. pp 48–55
30. Shih H-C (2017) A survey of content-aware video analysis for sports. *IEEE Trans Circuits Syst Video Technol* 28(5):1212–1231
31. Shi J, Tomasi (1994) Good features to track. In: IEEE Conference on Computer Vision and Pattern Recognition. pp 593–600



32. Sun S-W, Wang Y-CF, Huang F, Liao H-YM (2013) Moving foreground object detection via robust sift trajectories. *J Vis Commun Image Represent* 24(3):232–243
33. Sun M, Farhadi A, Seitz S (2014) Ranking domain-specific highlights by analyzing edited videos. In: *European Conference on Computer Vision*. pp 787–802
34. Tejero-de-Pablos A, Nakashima Y, Sato T, Yokoya N, Linna M, Rahtu E (2018) Summarization of user-generated sports video by using deep action recognition features. *IEEE Trans Multimedia* 20(8):2000–2011
35. Thành NT, Công PT et al (2019) An evaluation of pose estimation in video of traditional martial arts presentation. *Journal on Information Technologies & Communications* 2019(2):114–126
36. Tiwari V, Bhatnagar C (2021) A survey of recent work on video summarization: approaches and techniques. *Multimed Tools Appl* 80(18):27187–27221
37. Vasudevan V, Sellappa Gounder M (2021) Advances in sports video summarization—a review based on cricket videos. In: *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems*. pp 347–359
38. Voronina M (2019) Automated camera motion control for rhythmic gymnastics using deep learning. Master's thesis, Tallinn University of Technology, School of Information Technologies
39. Wei F, Wang B, Ge T, Jiang Y, Li W, Duan L (2022) Learning pixel-level distinctions for video highlight detection. In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp 3073–3082
40. Xu M, Wang H, Ni B, Zhu R, Sun Z, Wang C (2021) Cross-category video highlight detection via set-based learning. In: *IEEE/CVF International Conference on Computer Vision*. pp 7970–7979
41. Yan C, Li X, Li G (2021) A new action recognition framework for video highlights summarization in sporting events. In: *IEEE International Conference on Computer Science & Education*. pp 653–666
42. Zahan S, Hassan GM, Mian A (2023) Learning sparse temporal video mapping for action quality assessment in floor gymnastics. Preprint at <http://arxiv.org/abs/2301.06103>

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

## Authors and Affiliations

Marcos Rodrigo<sup>1</sup>  · Carlos Cuevas<sup>1</sup> · Daniel Berjón<sup>1</sup> · Narciso García<sup>1</sup>

Carlos Cuevas  
carlos.cuevas@upm.es

Daniel Berjón  
daniel.berjon@upm.es

Narciso García  
narciso.garcia@upm.es

<sup>1</sup> Grupo de Tratamiento de Imágenes (GTI), Information Processing and Telecommunications Center (IPTC), ETSI Telecomunicación, Universidad Politécnica de Madrid (UPM), Madrid 28040, Spain