

# AUTOMATIC PRODUCTION OF PERSONALIZED BASKETBALL VIDEO SUMMARIES FROM MULTI-SENSORED DATA

Fan Chen

School of Information Science  
Japan Advanced Institute of Science and Technology

Christophe De Vleeschouwer

Ecole Polytechnique de Louvain  
Université catholique de Louvain

## ABSTRACT

We propose a flexible framework for producing highly personalized basketball video summaries, by intergrating contextual information, narrative user preferences on story pattern, and general production principles. Starting from the multiple streams captured by a distributed set of fixed cameras, we study the implementation of autonomous viewpoint determination and automatic temporal segment selection, and also discuss the production of visually comfortable output, by applying smoothing process to viewpoint selection and by defining efficient benefit functions to evaluate various summary organization. The efficiency of our framework is demonstrated by experimental results.

**Index Terms**— Personalized Video Summarization, Content Repurposing, Viewpoint Selection

## 1. INTRODUCTION

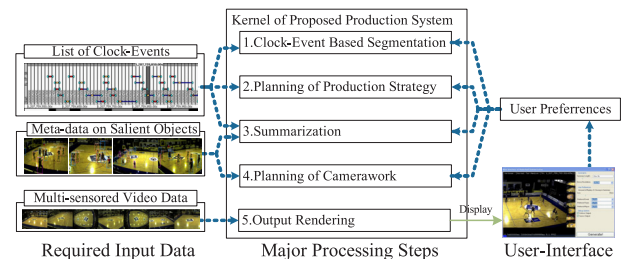
Towards increased user-centric adaptation of contents, democratic and personalized production of multimedia content is one of the most exciting challenges that content providers will have to face in the near future, to satisfy different user expectations concerning story-telling and heterogeneous terminal devices, in various applications ranging from interactive multimedia service [1][2] to intelligent surveillance [3]. Considering the problem in a multi-camera environment not only mitigates the difficulty of scene understanding caused by reflection, occlusion and shadow in the single view case, but also offers higher flexibility in producing visually pleasant video reports. In a typical application scenario, the sensor network for media acquisition is composed of (microphones and) cameras, which, for example, cover a basket-ball field. Distributed analysis and interpretation of the scene are exploited to decide what to show or not to show about the event, so as to produce a video composed of a valuable subset from the streams provided by each individual camera, or interpolated from multiple cameras.

In the present paper, we present a unified framework for cost-effective and autonomous generation of personalized video contents from multi-sensored data. Especially, we will focus on two issues: video production, which deals with

the automatic planning of camerawork to determine the viewpoint (including both the camera and the cropping area within this camera view) for rendering a scene; and video summarization, which determines the temporal part of the video to be presented to the audience. Due to page limitation, we omit the list of previous works, and invite readers to refer to a detailed review we made in [4]. Compared to those methods, our production system is able to deal with abundant semantic and narrative user preferences, and considers production principles for visually comfortable user experiences.

Based on our previous work on viewpoint selection [5] and on soccer video summarization[6], the present work provides two major improvements: 1) it combines automatic camerawork planning and summarization into a unified production system; 2) it improves the strategy for optimal viewpoint selection, both with respect to improved computational efficiency and behavior.

In Section 2, we will explain the proposed system in more details. Experimental data will be given in Section 3 to validate our work, before the concluding remarks in Section 4.



**Fig. 1.** Working flow (from top to bottom) of the proposed automatic production system of basketball summary.

## 2. PROPOSED PRODUCTION SYSTEM

Our key objective is to determine the proper part of content to present to the end-users, according to both user preferences and the contextual knowledge of the scene. Temporally, it acts as a summarization process through automatic clip selection. Spatially, it adaptively plans the camerawork to render the present scene with the optimal viewpoint. In Fig.1, we depict the overall framework of our production system. First, we divide a complete game into short clips, and then group consecutive clips into segments. As explained in Section 2.1, this segmentation is derived from (automatically generated)

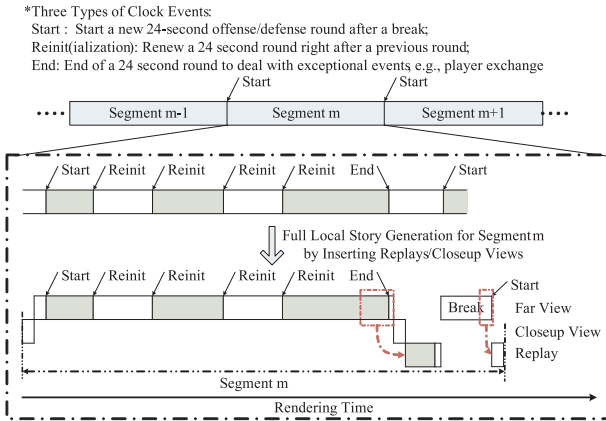
Part of this work has been funded by the Walloon Region WALCOMO project, by the FP7 APIDIS project, and by the Belgian NSF.

metadata that characterize the key events of the game. For each segment, we plan the production strategy by determining the view type and the position of replay insertion, based on a pre-defined rule. A story of the specified length is then organized from clips to meet user preferences, by solving a resource allocation problem. Finally, we perform camerawork planning on selected clips, and render the result out to the audiences.

Performing camerawork planning after summarization is meaningful because it saves the processing time on unselected clips, and provides full production personalization capabilities. For a large scale deployment, we may prefer to pre-encode the clips for several pre-specified camerawork options, to avoid computationally expensive online encoding.

### 2.1. Clock-event based video segmentation

According to the 24 seconds rule in basketball, the attacking team need to attempt a shot within 24 s of gaining possession. Many key events, including shooting, foul, interception and others, are closely related to starting/ending/restarting of clock counting, which are named "Start", "End" and "Reinit" clock event, respectively. Hence, it is natural for us to perform autonomous production in the divide-and-conquer paradigm for efficient processing, where clock events provide a reasonable base for video segmentation. Since most of critical actions (e.g. successful shots or fouls) lead to "End" clock event, it is safe and better to include all "Reinit" events in the same segment for a complete local story. We thus define a segment as the period between the clock time of two consecutive "Start" clock events, as shown in Fig.2.



**Fig. 2.** Based on clock events, we divide the video into segments, and plan the full local story of each segment, including both view-type determination and replay insertion.

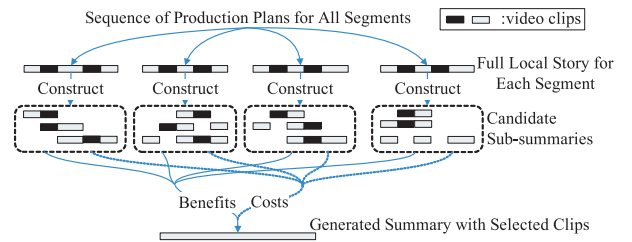
### 2.2. Production planning of full local story

Shots of various view types play an important role in telling an attractive story: far views are used to present the complexity of team sports, while closeup views are essential to increase the emotional involvement of audiences. For more important actions, replays should be appended for clearer explanation of local details[7]. In conventional sport video production, a director always has a rough plan in his mind on how to

organize those factors to present the game to viewers, based on general principles of sport video production. We simulate this process, and plan the production strategy for each segment before any computationally expensive optimization of clip and corresponding viewpoint selections, by only analyzing the structure of clock-events. In Fig.2, we take the following reference strategy to render each segment, i.e.,

- (1) A 2s close-up is taken 1 ~ 2s before the "Start" event.
- (2) A 1 ~ 2s close-up is inserted after the "End" event, depending on the length of break to the next "start" event.
- (3) A replay is inserted to cover a period from the last 1/3 part of the round before the "End" event to the starting point of the close-up in (2), if this period is longer than 3s.
- (4) We also insert a replay for the break between the end event and the next start event. Note that this break overlaps with the next segment, which we think is necessary to help audiences to reorient themselves in the new segment.
- (5) Other parts of the segment are rendered as far views.

After inserting replays/close-ups, we cut long sequences of frames belonging to the same shot type into shorter clips (~ 2s). It is worth mentioning here that the reference rendering strategy does not define the actual way a segment is rendered. Any subset of the clips of a segment actually defines an eligible rendering strategy. Selection of the optimal one is described in the next section.



**Fig. 3.** A resource allocation based framework of sport video summarization.

### 2.3. Resource allocation based summarization

This section considers selection of clips into the summary, given semantic preferences in terms of action or player, and narrative preferences in terms of story-telling style (replays or not, long or short segment stories). This process is based on the generic resource allocation framework we presented in ICIP2009 for soccer video summarization. Fig.3 briefly reminds the proposed summarization framework. For the  $m$ -th segment  $S_m$ , we consider  $L$  different narrative options  $\{s_{ml}\}$ , each option defining the subset of the clips of this segment that are rendered during display. A pair of cost/benefit values, i.e.,  $B(s_{ml})$  and  $C(s_{ml})$ , is assigned to each option  $s_{ml}$ , and a summary is obtained by maximizing the overall benefit under the length constraint  $u^{LEN}$ , i.e.,

$$\{s_{ml}^*\} = \arg \max_{\{s_{ml}\}} \sum_m B(s_{ml}), \quad \sum_m C(s_{ml}) \leq u^{LEN}, \quad (1)$$

which can be solved as a resource allocation problem by using Lagrangian Relaxation[8].

This summarization method allows highly personalized nonlinear story organization via flexible definition of benefits. In the present paper, the benefit is defined as

$$\mathcal{B}(s_{ml}) = \sum_{j \in s_{ml}} \mathcal{I}_{mlj} \mathcal{G}(s_{ml}, u^P, u^T) \mathcal{P}_{ml}^{CR}(u^C, u^R) \mathcal{P}_{ml}^F, \quad (2)$$

which includes semantical importance of clips  $\sum_{j \in s_{ml}} \mathcal{I}_{mlj}$  and extra gain  $\mathcal{G}(s_{ml}, u^P, u^T)$  from user favorite player  $u^P$  and team  $u^T$ , and also evaluates narrative preferences on story-telling (e.g., penalty  $\mathcal{P}_{mj}^{CR}$  on user specified story continuity  $u^C$ , and story redundancy  $u^R$ ). Satisfaction of general production principles is also evaluated through the penalty for forbidden cases  $\mathcal{P}_{mj}^F$ , to avoid frustrating visual/story-telling artifacts (e.g., over-short/incomplete local stories). Detailed explanations and definitions are available in [6].

#### 2.4. Autonomous planning of camerawork

Once clips have been selected, we construct their corresponding viewpoint sequence in a personalized way. We have improved both the search criterion and the strategy for viewpoint selection over our initial trials in [5]. Only the improvements compared to [5] will be explained in details. [5] has introduced three steps for automatic camerawork planning.

First, we determine the optimal viewpoint in each frame in each camera, which trades off between completeness (including more players in the selected viewpoint) and closeness (higher resolution on each individual player) for a specified display resolution. Formally, we define a viewpoint  $v_{ki}$  in the  $k$ -th camera view of the  $i$ -th frame, by its size  $S_{ki}$  and its center  $\mathbf{c}_{ki}$ . We assume that the position of players has been computed in each frame, e.g. using [9]. If there are  $N$  salient objects in this frame, and the location of the  $n$ -th object in the  $k$ -th view is denoted by  $\mathbf{x}_{nki}$ , we select the optimal viewpoint  $v_{ki}^*$  for far view and replay, by maximizing a weighted sum of object interests, i.e.,

$$\mathbf{v}_{ki}^* = \arg \max_{\mathbf{v}_{ki}} \beta(S_{ki}, \mathbf{u}) \sum_{n=1}^N I_n \alpha \left( \frac{\|\mathbf{x}_{nki} - \mathbf{c}_{ki}^{\text{SCN}}\|}{S_{ki}}, \mathbf{c}_{ki} \right). \quad (3)$$

In the above equation:

a)  $I_n$  denotes the importance assigned to the  $n$ -th object. It allows focusing on a preferred player by tuning its weight, depending on player identification, e.g. based on [9].

b) Function  $\alpha(\cdot)$  modulates the weights of the objects according to their distance to the scene center, normalized by the viewpoint size. This weight should be high and positive when the object-of-interest is within the viewpoint and close to the scene center  $\mathbf{c}_{ki}^{\text{SCN}}$ , and should be negative or zero when the object lies outside the viewing area.

Compared to the Mexican-Hat based implementation in [5] where  $\mathbf{c}_{ki}^{\text{SCN}}$  was set to  $\mathbf{c}_{ki}$ , here we set  $\mathbf{c}_{ki}^{\text{SCN}}$  to the ball position (or the gravity center of all objects when ball position is not available). Especially, we use the following  $\alpha(\cdot)$ , i.e.,

$$\alpha(\cdot) = \exp \left( - \frac{\|\mathbf{x}_{nki} - \mathbf{c}_{ki}^{\text{SCN}}\|^2}{2S_{ki}^2} \right) \times \mathcal{V}(\mathbf{x}_{nki}, S_{ki}, \mathbf{c}_{ki}), \quad (4)$$

where  $\mathcal{V}(\mathbf{x}_{nki}, S_{ki}, \mathbf{c}_{ki})$  is the visibility function, which takes 1 if object  $\mathbf{x}_{nki}$  is fully covered by viewpoint  $\mathbf{v}_{ki}$ , and  $-1$  for not. Since  $\mathcal{V}(\mathbf{x}_{nki}, S_{ki}, \mathbf{c}_{ki})$  can be easily computed by comparing the bounding box, this change significantly reduces the time for recalculating object interests when viewpoint moves.

c) Function  $\beta(\cdot)$  reflects the penalty induced when the native signal of the  $k$ -th camera has to be sub-sampled once the viewpoint size becomes larger than the maximal resolution  $u^{\text{res}}$  allowed by the user. An appropriate choice consists in setting the function equal to one when  $S_{ki} < u^{\text{res}}$ , and in making it decrease afterwards, e.g.,

$$\beta(\cdot) = \left[ \min \left( \frac{u^{\text{res}}}{S_{ki}}, 1 \right) \right]^{u^{\text{close}}}, \quad (5)$$

where  $u^{\text{close}} > 1$  increases to favor close viewpoints compared to large zoom-out views. Vector  $\mathbf{u}$  covers all these user preferences mentioned above, i.e.  $\mathbf{u} = [u^{\text{close}} u^{\text{res}} u^P u^T u^C u^R]$ .

We also reduce the size of solution space for searching optimal viewpoints. Eq.(3) reflects the trade-off between closeness and completeness. If we decay the  $\alpha(\cdot)$  slow enough, reducing closeness through virtual zooming out is only beneficial if it includes additional players in the viewpoint. In other words, it is useless to enlarge the viewpoint (thereby reducing closeness) without including any additional player. As a consequence, an optimal viewpoint will always be spanned by two players. We can thus restrict our initial full-search analysis to a selective search focusing on viewpoints of the required aspect ratio that include a pair of players on their border. Even an exhaustive selection of all possible pairs of (maximum 10) players can reduce computational complexity dramatically, compared to the grid search used in [5].

For close-up views, we simply find the minimal box that covers the closest player to the scene center  $\mathbf{c}_{ki}^{\text{SCN}}$ .<sup>1</sup> When the viewpoint is determined, we expand the viewpoint by 10% in all directions to leave a margin space for better appearance, as in conventional production[7].

As the second step, given the viewpoint selected in each camera view, we select the camera by comparing a criterion defined in terms of completeness, closeness and occlusion. Formally, the interest of selecting the  $k$ -th camera, i.e.,  $I_{ki}(\mathbf{v}_{ki}, \mathbf{u})$ , is defined as follows:

$$I_{ki}(\mathbf{v}_{ki}, \mathbf{u}) = w_k \beta(\cdot) \sum_{n=1}^N I_n h_k(\mathbf{x}_{nki}) \alpha(\cdot) o_k(\mathbf{x}_{nki}), \quad (6)$$

where  $w_k$  denotes the weight assigned to the  $k$ th camera, while  $\alpha(\cdot)$  and  $\beta(\cdot)$  are defined as in the first step above. Knowing the position of all other objects,  $o_k(\mathbf{x}_{nki})$  measures the occlusion ratio of the  $n$ -th object in camera view  $k$ , which is defined as the fraction of pixels of the object overlapped by

<sup>1</sup>Obviously it makes more sense to zoom on the player of interest (e.g. preferred player, or player who scored). We provide this as a provisional implementation before we have reliable automatic player identification, so as to complete the overall framework and leave space for future local revisions.

other objects when projected to the camera view.  $h_k(\mathbf{x}_{nki})$  is the height in pixels of projecting a six feet tall vertical object (average height of a player) located in  $\mathbf{x}_{nki}$  into camera view  $k$ , which serves as normalization of different camera views, and directly computed based on camera calibration. For far view, we assign higher weights to two major side-view cameras, while for replay higher weights are given to other cameras. Furthermore, for replay, we use  $o_k^2(\mathbf{x}_{nki})$  instead of  $o_k(\mathbf{x}_{nki})$  to emphasize more on reducing occlusions.

In the final step, an iterative smoothing process based on a two-layer Markov chain is applied to the selected sequences of viewpoints to remove visual artifacts such as flicking and fluctuation of the view (see [5]).

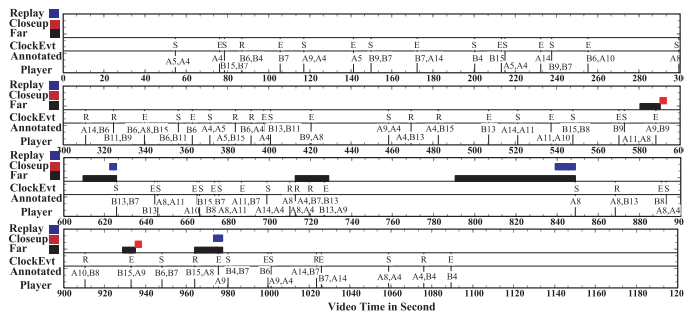


Fig. 4. A 2m30s summary generated with preference on player 9 of the yellow team (A9).

### 3. EXPERIMENTAL RESULTS

#### 3.1. Preparation of Meta-data

Two kinds of meta-data are required by the proposed system, i.e., salient objects and clock-events. Automatic detection of players and the ball were discussed in [9], where player identification was also briefly explored through player digit recognition. We have proved in [5] that our viewpoint selection method is robust against player miss-detection. For a real deployment, we can record the clocking signal by wiring into the clocking system, and recognize the type of events based on the relative locations of players and ball. Work is currently in progress within the APIDIS project(See [1]), and preliminary results have revealed that above 80% recognition rates are realistic. The more detailed the semantic information about the event (e.g., key events such as successful/failed shooting, free shot and its related foul can be accurately detected based on the information on current score and current possession team, which is recordable from the score board), the higher level of personalization can be offered to users.

We use automatically extracted salient objects and pre-recorded clock-events in the following experiments, while use manually annotated data of dominant players in each event.

#### 3.2. Results on automatic production of summaries

Due to page limitation, we only give one experimental result here, and invite reviewers to visit the demo page [10] for more experimental results and corresponding demo videos.

In Fig.4, we plot the status of clip selection for a 2min and 30sec summary generated with user preference on the player

9 of the yellow team, from a 14min game recorded by 7 cameras. In the bottom, we show the position of clock events (S for a starting event, R for a reinit event, and E for an ending event), and the annotated dominant players (A for the yellow team, and B for the black team). Note that annotations for the "End" clock event may cover both the clips before and after the end event, while annotations for a start/reinit event only apply to the clips after that event. We have the following observation from Fig.4.

1) By proper benefit definition, we suppress most of story-telling artifacts, such as standalone close-up/replays without the corresponding game parts.

2) Combined with the results in the demo page, our method could organize a summary which satisfies various user preferences on display resolutions, summary lengths, preferred players and so on. For example, when player A9 is preferred, our method could organize a story focusing on this player, with a proper story-telling pattern as shown in Fig.4.

### 4. CONCLUSION

We proposed a framework for producing personalized summaries of basketball videos from multi-sensored data. By taking divide-and-conquer strategy, we efficiently solve the problem of viewpoint determination and temporal segment selection. Especially, we defined the planning rule of production strategy and flexible criteria for viewpoint selection, and implemented a real-time production system. All these exploits a way to provide highly personalized video services to satisfy various user preferences, not only in basketball game, but also in many other application scenarios.

### 5. REFERENCES

- [1] Homepage of the APIDIS project.  
<http://www.apidis.org/>
- [2] Papaoulakis N. et al., "Real-time video analysis and personalized media streaming environments for large scale athletic events,"*AREA '08*, pp.105-112, 2008.
- [3] Yamasaki T., Nishioka Y., and Aizawa K., "Interactive retrieval for multi-camera surveillance systems featuring spatio-temporal summarization,"*MM '08*, pp.797-800,2008.
- [4] Chen F., Delannay D., De Vleeschouwer C., and Parisot P., "Multi-sensored vision for autonomous production of personalized video summary,"in "Computer Vision for Multimedia Applications: Methods and Solutions"(Eds. J. Wang et al.), *IGI Global*,(accepted), 2010.
- [5] Chen F. and De Vleeschouwer C., "Personalized production of team sport videos from multi-sensored data under limited display resolution,"*Computer Vision and Image Understanding*, vol.114, pp.667-680, 2010.
- [6] Chen F. and De Vleeschouwer C., "A resource allocation framework for summarizing team sport videos,"*ICIP 2009*, 2009.
- [7] Owens J., "Television sports production, 4th Ed.," *Focal Press*, 2007.
- [8] Everett H., "Generalized lagrange multiplier method for solving problems of optimum allocation of resources," *Operations Research*, vol.11 pp.399-417,1963.
- [9] Delannay D., Danhier N., and De Vleeschouwer C., "Detection and recognition of sports (wo)men from multiple views," *ICDSC'09*, 2009.
- [10] Demo Link:  
<http://www.jaist.ac.jp/~chen-fan/apidis/www/results-icip2010.html>