

# Be Your Own Cameraman: Real-Time Support for Zooming and Panning into Stored and Live Panoramic Video

Vamsidhar Reddy Gaddam<sup>1</sup>, Ragnar Langseth<sup>1</sup>, Håkon Kvale Stensland, Pierre Gurdjos<sup>2</sup>,  
Vincent Charvillat<sup>2</sup>, Carsten Griwodz<sup>1</sup>, Dag Johansen<sup>3</sup>, Pål Halvorsen<sup>1</sup>

<sup>1</sup>Simula Research Laboratory & University of Oslo, Norway  
{vamsidhg, ragnarla, haakonks, griff, paalh}@ifi.uio.no

<sup>2</sup>Universite de Toulouse, France  
{pgurdjos, vincent.charvillat}@enseeiht.fr

<sup>3</sup>Univerity of Tromsø, Norway  
dag@cs.uit.no

## ABSTRACT

High-resolution panoramic video with a wide field-of-view is popular in many contexts. However, in many examples, like surveillance and sports, it is often desirable to zoom and pan into the generated video. A challenge in this respect is real-time support, but in this demo, we present an end-to-end real-time panorama system with interactive zoom and panning. Our system installed at Alfheim stadium, a Norwegian premier league soccer team, generates a cylindrical panorama from five 2K cameras live where the perspective is corrected in real-time when presented to the client. This gives a better and more natural zoom compared to existing systems using perspective panoramas and zoom operations using plain crop. Our experimental results indicate that virtual views can be generated far below the frame-rate threshold, i.e., on a GPU, the processing requirement per frame is about 10 milliseconds.

The proposed demo lets participants interactively zoom and pan into stored panorama videos generated at Alfheim stadium and from a live 2-camera array on-site.

## Categories and Subject Descriptors

H.5.1 [Multimedia Information Systems]: Video; I.4.9 [Applications]: Video

## General Terms

Experimentation, Performance

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the Owner/Author.

Copyright is held by the owner/author(s).

*MMSys'14*, March 19-21, 2014, Singapore, Singapore  
ACM 978-1-4503-2705-3/14/03.  
<http://dx.doi.org/10.1145/2557642.2579370>

## Keywords

demonstration, panorama video, zoom, panning, real-time

## 1. INTRODUCTION

Panoramic video is becoming increasingly more popular in various scenarios. In the context of arena sports like soccer, it provides opportunities for innovation in broadcasting and in the area of game analysis. For example, based on a camera array covering the whole field, virtual views can be supported by zooming and panning into a stitched panorama video, e.g., allowing individual users to interactively control his or her own virtual camera. This type of challenges has attracted researchers' attentions for more than a decade [15, 17, 5, 7, 11, 6, 18], but often the zoom is made by simple cropping, or it does not scale. As can be seen in the literature, the challenges in providing such interactive experiences to a large audience span several fields, and one such challenge is to provide a real-time pannable and zoomable virtual camera to several thousands or even millions of users.

In this respect, we present an end-to-end real-time system to interactively zoom and pan into high-resolution panoramic videos. In contrast to many existing systems supporting zoom by cropping into a perspective panorama video, our system uses cylindrical panorama as an intermediate representation. Here, the perspective of the virtual view is corrected in real-time, and the result is a better and more natural zoom. The client has full freedom to pan, tilt and zoom into the panorama video using the system. The system supports (live switching between) several algorithms for pixel interpolation. Furthermore, our experimental results indicate that such zoomed virtual views can be generated far below the frame-rate threshold where an unoptimized GPU implementation generates a virtual view frame in about 10 ms. Thus, taking into account recent trends in device development, our approach should be able to scale to a large number of concurrent users in the near future to spectators and fans both within the stadium as well as outside. Inside the stadium, the Cisco connected stadium solution [4] already in use in more than 200 venues world-wide can be used to

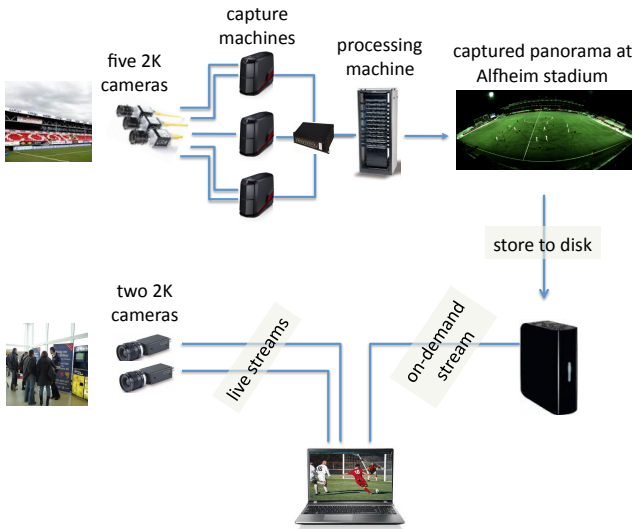


Figure 1: Demonstration setup

broadcast the panorama video to the spectators, and for users at home, there exist several scalable streaming solutions like segmented HTTP streaming [3, 13] or panorama video transfers in real-time [9].

In this demonstration, we present two versions of our system. The first is a demonstration which uses a recorded five 2K-camera panorama video from Alfheim stadium where the current prototype is installed. The second demonstration is a live, scaled down version<sup>1</sup> of the system with two 2K cameras zooming into live recorded video at the conference venue. The motivation is to provide a hands-on experience of such a system. The demo will allow users to interact with the virtual camera in both stored and live scenes.

## 2. SYSTEM OVERVIEW

The system presented in this demo is part of a larger system where the current prototype is running at Alfheim stadium in Tromsø. This system has been presented before [10, 14], but without the possibility to interactively zoom an pan into a high-resolution panorama.

### 2.1 System Setup

The relevant part of the system in this context is divided into a panorama generation part and a video delivery part, which supports our user-controlled interactive virtual cameras. A sketch of our system is given in figure 1. In this section, we describe these two parts.

#### 2.1.1 Real-time Panorama Video Capture

To capture the entire soccer field, we use five Basler industry vision cameras [2], each of which delivers a maximum resolution of  $2046 \times 1086$  pixels at 50 frames per second over

<sup>1</sup>The live 2-camera setup is due to hardware constraints bringing equipment to the conference venue. The five camera version requires several recording machines and a processing machine. A live view from the camera setup at Alfheim stadium could be possible, but as we sometimes have bandwidth problems over the 19-hop Tromsø-Oslo link, we expect an even lower available bandwidth from Tromsø to Singapore.

Gigabit Ethernet. We use an 8 mm lens [1] with virtually no image distortion, allowing us to bypass the lossy debarreling step of our previous real-time pipeline [16]. To maximize the panorama resolution, the cameras are rotated by  $90^\circ$  (see figure 2), giving a vertical field of view of  $66^\circ$ . Furthermore, the cameras are mounted in a circular pattern. This means that they are pitched, yawed and rolled to look directly through a point 5 cm in front of the lenses, in an attempt to reduce parallax effects. As a result of this, only minor adjustments are required before the images can be stitched together. The video capture system also determines the required exposure, which requires frequent changes due to quickly changing light conditions outdoors. In this respect, auto-exposure is performed on the center camera once every 4 seconds, and the camera reader module broadcasts the resulting exposure parameters to the other camera readers.



Figure 2: Mounted camera array

After the generation of the panorama video, it is encoded and compressed using x264. At the moment, with a focus on video quality, the system still demands a large network bandwidth to the full resolution panorama video. However, there are large potentials for trading of quality for lower bandwidth requirements (but an investigation of this is out of scope for this demonstration).

#### 2.1.2 Live Panoramic Zoom and Panning

To deliver a virtual camera to clients, the panorama video is delivered to the client device. Currently, we use HTTP segment streaming (no quality adaption at the time of writing, but it can easily be added in the future). The segments of the panoramic videos are served by an Apache server along with a manifest file. The manifest file is used to inform the clients when the next file is ready for download. In case of live streaming, the viewer checks the manifest file periodically and downloads the next segment when ready. As soon as the panoramic video segment is transferred, it is decoded for processing by the multi-threaded client process. The downloading thread runs in the background without blocking either the display thread or the user input thread.

Using the cylindrical panorama as an intermediate representation, our system then generate an arbitrary virtual camera view from the position of the camera array. All the details of the pixel calculations are out of scope of this demonstration description (see [8]), but the main operation is to find the pixels of the virtual camera image from the cylindrical texture as shown in figure 3. A pin-hole camera model is used, and the a point projection from a 3D point to an image point is calculated using a 3D rotation matrix as a function of the rotation angles around the x, y and z axes. A particular pixel is then found by calculating the ray that passes from the camera center to the pixel. The intersection of this ray with the cylinder gives the exact position on the cylindrical texture. Finally, since there is sub-pixel accu-

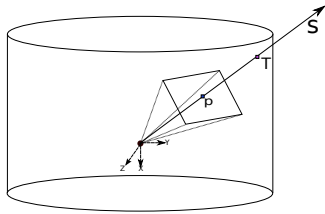


Figure 3: The intersection of the ray from the virtual view with the unit cylinder of the panorama image.

racy of the intersection calculation, the used pixel value is calculated using interpolation of all the surrounding pixels.

The user has the full freedom to pan, tilt and zoom, being his or hers own camera-man<sup>2</sup>. The virtual camera view is corrected to a perspective view very similar to that of a physical camera as shown in figure 4.

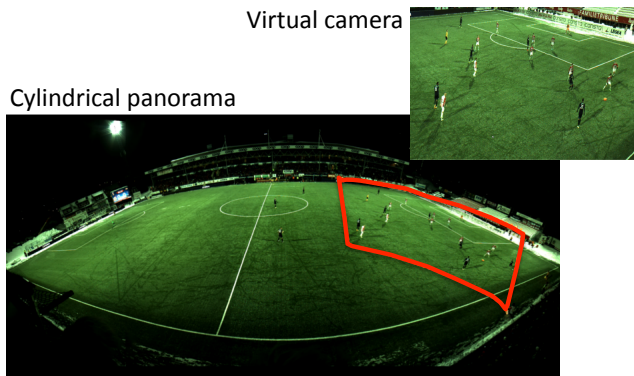


Figure 4: The virtual camera is generated from the region of interest marked in the panorama video. Note that it is not a simple crop from the bigger video.

## 2.2 System Performance

We have earlier shown that our system can generate panorama video in real-time using four 1K cameras [16]. Going for higher resolution, we have now distributed the capture and processing of five 2K cameras, and we are able to generate the panorama frames in 34.58 ms on an Intel i7 CPU. This is below our 25 fps threshold. The generated video can either be stored to disk or forwarded directly to connected client devices.

When presented to the user, the zooming and panning operations can be performed on both the sender and receiver side. In the current setup, we as explained above transmit the panorama video to the display device like a laptop. Here, the zoom and pan operations are executed where the perspective is corrected from the cylindrical panorama in real-time. We use the GPU on the client side, and each frame for virtual camera view with Full HD resolution ( $1920 \times 1080$ ) can be generated in about 10 milliseconds on a machine with an Intel i7-2600 CPU and an Nvidia GeForce GTX 460 GPU

<sup>2</sup>An example video using the known ball position, where the virtual camera-man is following the ball is shown here: <http://www.youtube.com/watch?v=554RjEEtw3o>

– again, far below the target 25 fps requirement. This time is dependent of the output resolution, and going for a mobile phone resolution it can be further reduced by a millisecond or two, e.g., about 1.5 milliseconds for iPhone 4 and 5.

## 3. DEMONSTRATION

In this demo, we present a system for real-time interactive zooming and panning of panorama video. The general setup is shown in figure 1. We use two types of panorama input. We use stored panorama video captured in the Alfheim soccer stadium (the European League game between Tromsø IL and Tottenham Hotspurs) [12]. The video was recorded using five 2K industrial cameras, and processed and stitched in a distributed system. We also use a live system using 2 cameras on site to generate the panorama video. Both the stored-video and the live-video demos work in real-time, and the user interactions have a very low delay from command to visual response – thus, a user can be his or her own cameraman!

## Acknowledgments

This work has been performed in the context of the *iAD* center for Research-based Innovation (project number 174867) funded by the Norwegian Research Council. Furthermore, the authors also acknowledge the support and hardware given by Hugo Kohmann and Roy Nordstrøm in Dolphin Interconnect Solutions.

## 4. REFERENCES

- [1] Azure-0814m5m. <http://www.azurephotonicsus.com/products/azure-0814M5M.html>.
- [2] Basler aca2000-50gc. <http://www.baslerweb.com/products/ace.html?model=173>.
- [3] SmoothHD. <http://www.smoothhd.com>, 2009.
- [4] Sports & Entertainment – Cisco Connected Stadium, 2013. <http://www.cisco.com/web/strategy/sports/index.html>.
- [5] Y. Arik, S. Kubota, and M. Kumano. Automatic production system of soccer sports video by digital camera work based on situation recognition. In *Proc. of ISM*, pages 851–860, 2006.
- [6] P. Carr and R. Hartley. Portable multi-megapixel camera with real-time recording and playback. In *Proc. of DICTA*, pages 74–80, 2009.
- [7] P. Carr, M. Mistry, and I. Matthews. Hybrid robotic/virtual pan-tilt-zoom cameras for autonomous event recording. In *Proc. of ACM MM*, pages 193–202, 2013.
- [8] V. R. Gaddam, R. Langseth, S. Ljødal, P. Gurdjos, V. Charvillat, C. Griwodz, and P. Halvorsen. Interactive zoom and panning from live panoramic video. In *Proc. of ACM NOSSDAV*, Mar. 2014.
- [9] R. Guntur and W. T. Ooi. On tile assignment for region-of-interest video streaming in a wireless lan. In *Proc. of NOSSDAV*, pages 59–64, 2012.
- [10] P. Halvorsen, S. Sægrov, A. Mortensen, D. K. Kristensen, A. Eichhorn, M. Stenhaus, S. Dahl, H. K. Stensland, V. R. Gaddam, C. Griwodz, and D. Johansen. Bagadus: An integrated system for

- arena sports analytics – a soccer case study. In *Proc. of ACM MMSys*, pages 48–59, Mar. 2013.
- [11] A. Mavlankar and B. Girod. Video streaming with interactive pan/tilt/zoom. In M. Mrak, M. Grgic, and M. Kunt, editors, *High-Quality Visual Experience*, Signals and Communication Technology, pages 431–455. Springer Berlin Heidelberg, 2010.
- [12] S. A. Pettersen, D. Johansen, H. Johansen, V. Berg-Johansen, V. R. Gaddam, A. Mortensen, R. Langseth, C. Griwodz, H. K. Stensland, and P. Halvorsen. Soccer video and player position dataset. In *Proc. of ACM MMSYS*, Mar. 2014.
- [13] R. Pantos (ed). HTTP Live Streaming. <http://www.ietf.org/internet-drafts/draft-pantos-http-live-streaming-10.txt>, 2013.
- [14] H. K. Stensland, V. R. Gaddam, M. Tennøe, E. Helgedagsrud, M. Næss, H. K. Alstad, A. Mortensen, R. Langseth, S. Ljødal, O. Landsverk, C. Griwodz, P. Halvorsen, M. Stenhaug, and D. Johansen. Bagadus: An integrated real-time system for soccer analytics. *ACM Trans. Multimedia Comput. Commun. Appl. (TOMCCAP)*, 10(1s):14:1–14:21, Jan. 2014.
- [15] X. Sun, J. Foote, D. Kimber, and B. Manjunath. Region of interest extraction and virtual camera control based on panoramic video capturing. *IEEE Transactions on Multimedia*, 7(5):981–990, 2005.
- [16] M. Tennøe, E. Helgedagsrud, M. Næss, H. K. Alstad, H. K. Stensland, V. R. Gaddam, D. Johansen, C. Griwodz, and P. Halvorsen. Efficient implementation and processing of a real-time panorama video pipeline. In *Proc. of IEEE ISM*, Dec. 2013.
- [17] J. Wang, C. Xu, E. Chng, K. Wah, and Q. Tian. Automatic replay generation for soccer video broadcasting. In *Proc. of ACM MM*, pages 32–39, 2004.
- [18] M. Wieland, R. Steinmetz, and P. Sander. Remote camera control in a distributed multimedia system. In B. Wolfinger, editor, *Innovationen bei Rechen- und Kommunikationssystemen*, Informatik aktuell, pages 174–181. Springer Berlin Heidelberg, 1994.