
Experiencing the Ball's POV for Ballistic Sports

Kodai Horita

UEC Tokyo
1-5-1 Chofugaoka, Chofu
Tokyo, Japan
horita@vogue.is.uec.ac.jp

Hideki Sasaki

UEC Tokyo
1-5-1 Chofugaoka, Chofu
Tokyo, Japan
sasaki@vogue.is.uec.ac.jp

Kris M.Kitani

Carnegie Mellon University
5000 Forbes Ave.
Pittsburgh, PA. USA
kkitani@cs.cmu.edu

Hideki Koike

UEC Tokyo
1-5-1 Chofugaoka, Chofu
Tokyo, Japan
koike@vogue.is.uec.ac.jp



Copyright is held by the author/owner(s).
CHI'13, April 27 – May 2, 2013, Paris, France.
ACM 978-1-XXXX-XXXX-X/XX/XX.

Abstract

We place a small wireless camera inside an American football to capture the ball's point-of-view during flight to augment a spectator's experience of the game of football. To this end, we propose a robust video synthesis algorithm that leverages the unique constraints of fast spinning cameras to obtain a stabilized bird's eye point-of-view video clip. Our algorithm uses a coarse-to-fine image homography computation technique to roughly register images. We then optimize an energy function defined over pixel-wise color similarity and distance to image borders, to find optimal image seams to create panoramic composite images. Our results show that we can generate realistic videos from a camera spinning at speeds of up to 600 RPM.

Author Keywords

Digital Sports, BallCam, Video Synthesis, Image Stitching.

ACM Classification Keywords

H.5.m [H.5.1 Information Interfaces and Presentation: Artificial, Augmented, and Virtual realities.]: Miscellaneous.

Introduction

We propose a novel viewing experience for ball-based sports games by presenting a framework for generating



Figure 1: Our BallCam prototype allows spectators to experience sports from a ball's point-of-view.

stabilized videos from the ball's point-of-view(POV). Our prototype ball-camera system is composed of a single wireless camera embedded in an American football. Using our ball-camera system, a spectator is able to experience the flight of the ball and being caught by the receiver (figure 1). We extend the original method of Kitani *et al.* [6] by introducing a more robust coarse-to-fine image homography computation technique and a graph-based image stitching technique to handle more complex configurations of image overlap.

Recent advances digital device technology permit them to be become small. We can put these small device(camera, accelerometer etc..) into the sports ball, and the ball can augment ball-based sports.

Some ball-type device are already developed. SHOOTBALL[9] suggested a new type interaction sports by using ball-type device and screen. This system is consist of shock sensor,wireless module and battery into the ball. The shock sensor can analyze of timing of bounce and the wireless module sends the information to the game server. Bouncing Star[5] is interactive media art project of a smart-ball system. This smart-ball consist of some sensor(accelerometer, microphone,wireless module

etc..). The system recognize ball's position,bouncing, spinning etc.. and give some output(Computer Graphic, music etc..). These projects show that Digital sports give us new type ball-based entertainment sports. But Digital sports is not only entertainment. Recently, ball-type device is used as new type judgement system.Soccer's "Goal Line Technology(GLT)" is the most popular.GLT has two system. One is the "Hawk-eye system[4]" which uses some camera and another is the "GoalRef[1]" . "GoalRef" consists of coils in ball and magnetic field is generated around the goal.When the ball past over the line, software can detect the condition of the magnetic field in the goal changed.

H. Mori *et al.* [7] developed a ball camera system using multiple cameras to generate a stabilized video. Their system used an optical flow-based approach to estimate camera rotation parameters. Pfeil *et al.* [8] introduced a system to capture a single static spherical panorama and triggers the shutter using an accelerometer. In contrast to past systems, our prototype works with only a single camera and is able to deal with extremely fast camera motion(600 RPM).

Generating the ball's point of view

Since a typical football rotates at roughly 600 RPM while in flight, the high speed movement of the camera poses several significant challenges that make the task of extracting a stabilized downward looking video very difficult. In this section, we explain how, under certain domain assumption and by leveraging the unique characteristics of camera images recorded under high speed rotation, we can create plausible ball POV videos in the context of American football.

Removing Camera Distortion

We begin by removing the barrel distortion of the lens and the rolling shutter distortion caused by the camera motion and the CMOS sensor image acquisition latency. If a camera has no lens distortion, an image point distorted by the rolling-shutter effect x_d and its rectified position x_r are related by the following equation [3]:

$$x_r = KR^\top(t)K^{-1}x_d \quad (1)$$

where K is a calibration matrix of the camera and $R(t)$ is a camera rotation. Since the camera is moving, the rotation depends on time t , and t is proportional to the image row.

In order to correct the lens distortion and the rolling-shutter distortion simultaneously, we extend (1) as:

$$x_r = KR^\top(t)\mathcal{P}^{-1}(x_d) \quad (2)$$

where \mathcal{P}^{-1} represents the back-projection from the image plane to 3D space. Note that this back-projection function takes the lens distortion into account. We use a simple radial distortion model in our implementation [2].



Figure 2: Result of image rectification.

To perform image rectification, we need to compute the distorted image point x_d for each rectified image pixel position x_r . We use the Gauss-Newton method to solve (2) for x_d and create a lookup table for image rectification. This process is illustrated in Figure 2.

View Expansion

Following [6] we use the mean intensity of the images to first generate a sequence of images that share a similar viewing angle. However, simply interpolating between sub-sampled image results in a very shaky video since the axis of rotation is not perfectly orthogonal to the camera axis and the camera rotation is not in sync with the camera frame rate. Before we proceed to synthesize novel

views between images to temporally up-sample the video, we would like to expand the field-of-view for each image frame. This is needed because there is often large displacements between subsampled frames and we must ensure we can interpolate between images without ‘holes’ in the image frames. We do this by first generating small composite images using sets of 3 temporally neighboring images. Next, we further expand the triplet image by using neighboring triplet images from adjacent rotation cycles (Figure 3).



Figure 3: View expansion

To generate composite images, it is necessary to compute the image transformation between images by computing

their homographies. However, since the football field has many repetitive patterns and the image distortion makes feature matching very challenging, we introduce a robust coarse-to-fine homography estimation technique to ensure reliable image registration.

In the coarse step, we assume an affine motion model that only allows for translational motion A , by solving the following linear equation,

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = A \begin{bmatrix} x \\ y \end{bmatrix}, \quad (3)$$

where x and y are the point of feature point before frame image, x' and y' are the point of feature point of next frame. We can solve for A via linear regression and then proceed to remove outlier points that have high variance. With the remaining points we use RANSAC to estimate a full homography matrix H to account for changes in perspective and scaling.

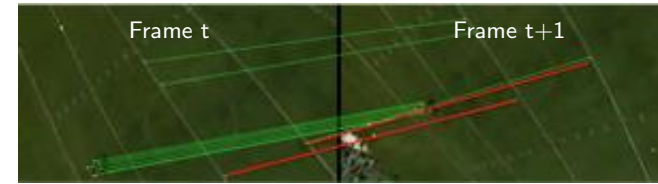


Figure 4: Bad point correspondences (marked in red) identified via a coarse-to-fine motion estimation

Simply blending the composite images using a homography still causes significant image noise (i.e. the world is not static and planar). To make our method robust to such noise, we pose the image stitching problems as a graph cutting problem.

We use the pairwise color continuity cost between two pixels i and j , $Cc(i, j)$, which is computed as the color

difference between the foreground image F and the background image B ,

$$C_{\text{color}}(i, j) = \frac{1}{N} \left(\|F(i) - B(i)\|^2 + \|F(j) - B(j)\|^2 \right), \quad (4)$$

where $F(\cdot)$ and $B(\cdot)$ are vectors of RGB values.

We also introduce a distance potential that takes into account the distance to an image boundary,

$$C_{\text{dist}}(i, j) = \min\{\dots\} \quad (5)$$

where distance function compute the minimum distance to a boundary pixel (regions of no image overlap).

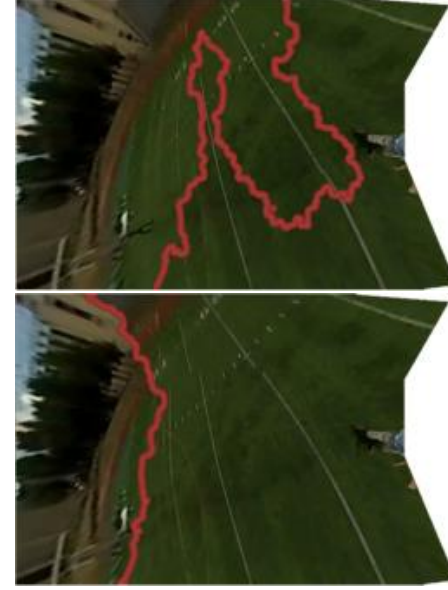
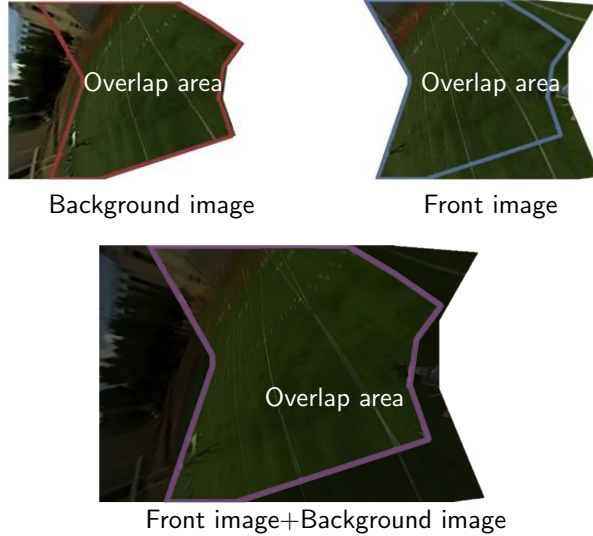


Figure 5: Graphut with color cost, with color and distance

Generating a Virtual Path

To generate a virtual camera path we compute a series of image warps based on the computed homographies between composite images. From a homography we can compute a per-pixel mapping $F(x, y)$ that describes how a single pixel in frame t can be transformed into a pixel in frame $t + 1$. We can generate an arbitrary view between these two frames using linear interpolation between an identity mapping I and the forward mapping F or backward mapping B , for frame t and $t + 1$, respectively,

$$F'(x, y) = (1 - \alpha)I(x, y) + \alpha F(x, y), \quad (6)$$

$$B'(x, y) = (\alpha)I(x, y) + (1 - \alpha)B(x, y). \quad (7)$$

However, this first-order interpolation introduces a high frequency motion component as an artifact of aliasing

(i.e. the synthesized view sways from side to side).

In order to account for this camera motion noise, we use a second-order interpolation by interpolating between half maps.

This method is based on the simple idea. In the first-order interpolation case, when we translate the frame t image to $t + 1$ image, we must generate all interpolate image sequences between t and $t + 1$. However, if we start the half frame $t + 0.5$ image between t and $t + 1$, we only generate the image sequence between $t + 0.5$ and $t + 1$. In the second-order interpolation case, we set the start point and end point of the interpolation to half frame between the images. This interpolation method generate the intermediate image from mapping before beginning of linear interpolation. We can generate the image of the half maps by using a per-pixel mapping F, B and I .

$$InF(x, y) = I(x, y) \times 0.5 + F(x, y) \times 0.5, \quad (8)$$

$$InB(x, y) = I(x, y) \times 0.5 + B(x, y) \times 0.5. \quad (9)$$

We must generate the half map image before linear motion interpolation. And then, we compute the homography and maps of half map images.

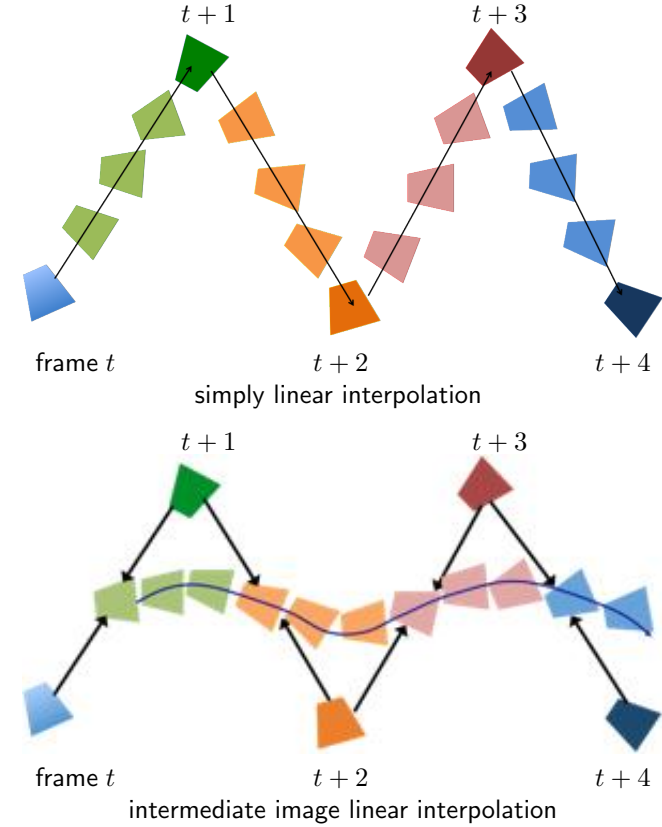


Figure 6: linear interpolation and intermediate image linear interpolation

Discussion

In this section, we discussed the result and our future work.

Remove distortion

Because our ball camera system is spinning so quickly, the video is always distorted by motion blur and rolling

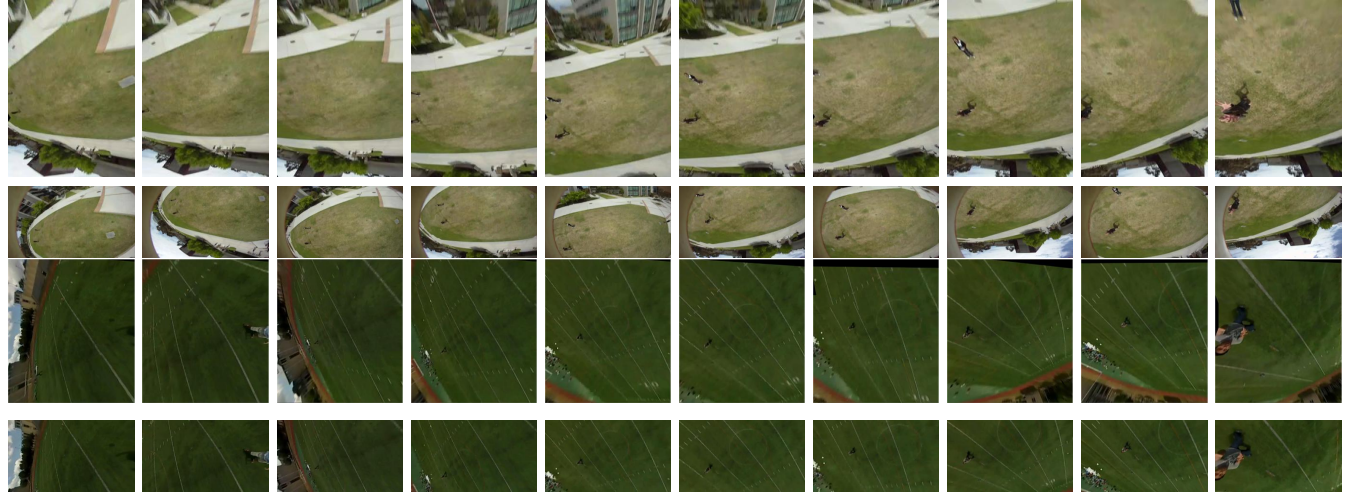


Figure 7: BallCam Results: (top) our approach, (bottom) naive sub-sampling

shutter effect. To remove the distortion, we developed the system of remove distortion. Figure 2 is the result. The result shows our system can remove distortion of spinning camera.

Image stitching by graph cut algorithm

To expand the ball camera's view, we implemented the image stitching algorithm. First, we used graph cut with only color continuity cost. But, we couldn't have the satisfaction of this result. It was because the next frame image encroached on present frame image area. To solve this problem, we improved the graph cut algorithm with color continuity cost and distance cost. This algorithm could select the cut line near the next frame image and get the optimal results.

Result movie

Figure 7 shows the result of ball's point-of-view sequences by using our system. The system enables generating ball's point-of-view video from the spinning camera. And then, our system can generate the video which taken by different fps camera. Top of the figure 7 is the result taken by 60 fps video and the bottom result taken by the 120 fps camera. But, regardless of we remove distortion of the video before generating the video, the distortion sometimes have an implication for the result video.

futurework

Our project's goal is that we will use the our ball camera system in the real ball-based sports game. This system has some improved points. First, we will try to put a camera into different type ball. In this paper, we tried to throw american football, because this balls rotation axis is

more stationary than other balls. But we think that our system is used by ball-based sports. It is not only american football. Second, we will solve weather condition issue. Real ball-based sports games are not necessarily fine weather. But our system is only work at fine weather condition. It is because our system use image intensity. Thus, we must estimate camera direction by other approach. Now, our system need time about 10 15 minutes to generate the one video and the result video is low image quality.

Conclusion

We have proposed a generating novel balls point of view video sequence from a spinning camera. Our system shows a new type of spectator technology for ball-based sports. we implement the two linear interpolations, but now we try to implement new interpolation method. In the future, we may generate more smoothly video by the interpolation method.

References

- [1] fraunhofer. Goalref. <http://www.iis.fraunhofer.de/en/bf/ln/referenzprojekte/goalref.html>.
- [2] Hartley, R., and Zisserman, A. *Multiple View Geometry in Computer Vision*, second ed. Cambridge University Press, 2004.
- [3] Hedborg, J., Ringaby, E., Forssen, P.-E., and Felsberg, M. Structure and motion estimation from rolling shutter video. In *Computer Vision Workshops (ICCV Workshops)* (2011), 17–23.
- [4] Innovations, H.-E. Hawk-eye. <http://www.hawkeyeinnovations.co.uk/>.
- [5] Izuta, O., Sato, T., Kodama, S., and Koike, H. Bouncing star project: design and development of augmented sports application using a ball including electronic and wireless modules. In *Proceedings of the 1st Augmented Human International Conference, AH '10*, ACM (New York, NY, USA, 2010), 22:1–22:7.
- [6] Kitani, K., Horita, K., and Koike, H. Ballcam!: dynamic view synthesis from spinning cameras. In *Adjunct proceedings of the 25th annual ACM symposium on User interface software and technology, UIST Adjunct Proceedings '12*, ACM (New York, NY, USA, 2012), 87–88.
- [7] Mori, H., Sekiguchi, D., Kuwashima, S., Inami, M., and Matsuno, F. Motionsphere. In *ACM SIGGRAPH 2005 Emerging technologies, SIGGRAPH '05*, ACM (New York, NY, USA, 2005).
- [8] Pfeil, J., Hildebrand, K., Gremzow, C., Bickel, B., and Alexa, M. Throwable panoramic ball camera. In *SIGGRAPH Asia 2011 Emerging Technologies, SA '11*, ACM (New York, NY, USA, 2011), 4:1–4:1.
- [9] Sugano, Y., Ohtsuji, J., Usui, T., Mochizuki, Y., and Okude, N. Shootball: the tangible ball sport in ubiquitous computing. In *Proceedings of the 2006 ACM SIGCHI international conference on Advances in computer entertainment technology, ACE '06*, ACM (New York, NY, USA, 2006).