## Ball Position and Motion Reconstruction from Blur in a Single Perspective Image

Giacomo Boracchi

Vincenzo Caglioti

Alessandro Giusti

Politecnico di Milano

Dipartimento di Elettronica e Informazione

E-mail: {giacomo.boracchi|vincenzo.caglioti|alessandro.giusti}@polimi.it

#### Abstract

We consider the problem of localizing a moving ball from a single calibrated perspective image; after showing that ordinary algorithms fail in analyzing motion blurred scenes, we describe a theoretically-sound model for the blurred image of a ball. Then, we present an algorithm capable of recovering both the ball 3D position and its velocity. The algorithm is experimentally validated both on real and synthetic images.

### 1. Introduction

Basic projective geometry allows to reconstruct the 3D position of a sphere from a single perspective image, provided that the camera parameters and the sphere radius are known. However, this is rarely useful in practical applications where moving ball is captured: in fact, as the exposure time is not infinitesimal in physical imaging devices (both video and photo cameras), the moving ball rarely projects to a crisp ellipse in the image. As a matter of fact, it often appears as an elonged smear without sharp contours, confusing most computer vision algorithms (see figure 1). The straightforward approach for determining the sphere 3D position by fitting an ellipse in a single calibrated image fails in this scenario.

In this work we present a novel algorithm to estimate the 3D position and velocity of a uniformly-colored moving ball from a single image, by explicitly analyzing and exploiting motion blur. Contrarily to most related works, we use a realistic image formation model to handle perspective images, where blur is not uniform.

In [4], we adopted a similar approach for reconstructing the curvilinear trajectory of a ball from a single longexposure image. Here we consider the complimentary case, in which the exposure is short enough to encompass only a short part of the ball trajectory. This characterizes most ordinary photographs and video frames, and also allows us to assume a rectilinear, uniform-speed ball motion during the exposure.

There are several works in literature that deal with motion blur; most of them ([1, 6]) aim at removing blur artifacts ("deblurring"), which greatly improves visual effectiveness of images and performance of object classification algorithms. However, this hardly permits exact measurements needed for 3D reconstruction. Motion estimation techniques from a single blurred image have been proposed in [14, 15] to estimate the speed of a moving vehicle and of a moving ball, respectively. However these works assume uniform blur as perspective is neglected.

Ball localization and tracking in videos is the object of many works, such as [8, 10, 11], where the trajectory of a moving ball is reconstructed by tracking the ball through the frames of a video sequence. However, these approaches require that the ball is visible from multiple synchronized cameras, whose corresponding frames are analyzed in order to triangulate the ball's position. Motion blur is neglected, as the ball contours are never used. Reid in [17] provides a method to reconstruct the ball position and motion from a single video sequence by analyzing its shadow: this must be visible and recognizable – which is a rather strict requirement in many application scenarios. In [13, 16], a physics-based approach is proposed, to estimate the parameters of a parabolic trajectory.

The considered image model is presented in Section 2 while the algorithm is described in Section 3. In Section 4 experiments are presented and discussed; Section 5 delineates ongoing and future works.

## 2. Blurred ball image model

A motion blurred image Z occurs whenever the scene projection on the image plane changes during the camera exposure period  $e = [t_0, t_0 + \Delta t]$ . Then we model Z as the integration of infinite sharp images, each exposed for an infinitesimal portion of e. Equivalently, Z can be considered as the temporal average of infinite sharp images  $I_t$ , each taken with the same exposure time  $\Delta t$  and representing the scene frozen at a different instant  $t \in e$  (see figure 2). This



Figure 1. Top: example motion blurred ball images; the rightmost one results from a very long exposure, and is not handled by our described technique. Bottom: an image (note perspective distortion), gradient intensity and unsatisfying results of canny operator.

can be formalized as

$$Z(x) = \frac{\int_{e} I_t(x)dt}{\Delta t} + \eta(x), \tag{1}$$

where x represents pixel coordinates ranging on a discrete grid X and  $\eta \sim N(0, \sigma)$  represents additive gaussian white noise. Many 3D rendering packages exploit the model (1) for accurate synthesis of motion blurred images.

We further assume that Z depicts a moving ball, and that every image  $I_t$  shows a uniformly colored ellipse<sup>1</sup> over a uniform background.  $I_{t_0}$  and  $I_{t_0+\Delta t}$ , respectively represent the ball at the beginning ("first curtain") and at the end ("second curtain") of the exposure. Let  $c_1$  and  $c_2$  be the ellipses appearing in these two images: these are used for 3D sphere motion estimation, as they represent the sphere at time  $t_0$  and  $t_0 + \Delta t$ . Figure 2 represents  $c_1$  and  $c_2$  and other ellipses taken from some images  $I_t$  in between. Since the ball trajectory is linear and the speed is uniform, all the ellipses have two tangent lines in common, which converge to the vanishing point v associated to the sphere displacement direction.

Because of perspective, the  $I_t$  images are not related by a simple translation: therefore, model (1) can not be expressed as a convolution with a point spread function, being more general.

In the following, we will ignore the contribution of the image background and ball shading and pigmentation, by assuming that in every  $I_t$  the ball image has a uniform color over a roughly uniform background. Some techniques ([4]) allow to drop this restrictive assumption, under reasonable constraints on the ball surface colors and provided that the

background is known. In short, we reduce to the case where the intensity of each pixel is directly proportional to the time the ball image covers that pixel. In Section 4, we show that our technique performs reasonably well also in real images where these assumptions are not precisely met.

Note that contours  $c_1$  and  $c_2$  are hardly recognizable from the ball smear, as they are not characterized by a discontinuity in image intensity. On the contrary, the rectilinear bounds (bitangent to  $c_1$  and  $c_2$ ) are more visible, although they are not ordinary "step" contours either (refer to [4]).

### **3** Reconstruction technique

Our reconstruction technique exploits the blurred image model, described in Section 2, in order to recover the ellipses  $c_1$  and  $c_2$ .

At the beginning, we roughly estimate the projection of the ball motion direction on the image plane; then, analysis of intensity profiles along this direction determines points lying on  $c_1$  and  $c_2$ . Finally,  $c_1$  and  $c_2$  are estimated considering some geometrical constraints; this allows us to reconstruct the position of the ball at the beginning and end of the exposure.

### 3.1 Blur direction estimation

In orthographic images the uniform blur assumption holds, and the motion blur direction can be estimated as the direction minimizing  $\ell^2$  norm of directional derivatives of

<sup>&</sup>lt;sup>1</sup>A sphere in 3D space appears as an ellipse in a perspective image.



Figure 2. Model of a motion blurred ball: the blurred image can be interpreted as a temporal average of multiple images, in each of which the ball is still, at a different position. Note that due to perspective the two tangents to all the ellipses are not parallel.

Z ([18]):

$$\widehat{\theta} = \arg\min_{\theta} \sum_{\mathbf{x} \in X} \left\| D_{\theta} (Z) (\mathbf{x}) \right\|^2,$$
(2)

where  $D_{\theta}$  can be any derivative filter along direction  $\theta$ . The minimization of (2) can be done in a closed form using steerable filters [7].

However in perspective images there is no unique blur direction, as blur is space varying and it is always directed towards the vanishing point v. If eccentricities of  $c_1$  and  $c_2$  are small compared to the blur extent, the blur directions are close to symmetric w.r.t. to the projection of the 3D ball displacement;  $\hat{\theta}$  represents the projection of the ball displacement.

Therefore  $\hat{\theta}$  is used to initialize the algorithm both on perspective and orthographic blurred images.

## **3.2** Finding points on $c_1$ and $c_2$

The procedure used to extract  $c_1$  and  $c_2$  exploits the pixel intensity values along a line ("profile"), having direction  $\hat{\theta}$ . We separately consider n profiles  $\{p_i(x)\}_{i=1,..,n}$ .

As shown in Figure 3, each profile p intersects  $c_1$  at two points ( $B_1$  and  $F_1$ ), and  $c_2$  at  $B_2$  and  $F_2$ . Profiles not intersecting both  $c_1$  and  $c_2$  are ignored in the current implementation.

On each profile, we identify five segments. The first and last  $([O, B_1] \text{ and } [B_2, E])$  contain pixels never covered by the moving ball; they are therefore characterized by the constant background intensity b. The neighboring segments  $[B_1, F_1]$  and  $[F_2, B_2]$  contain pixels which have been covered by the ball only during part of the exposure: intensity values at these pixels are linearly increasing (decreasing) along the profile as the ball moves at uniform speed<sup>2</sup>. The central part  $[F_1, F_2]$  contains pixels covered by the ball during the whole exposure; their value corresponds to the ball intensity f.

We localize points  $B_1$ ,  $B_2$ ,  $F_1$  and  $F_2$  along profiles as these determine  $c_1$  and  $c_2$ . Because of noise, shading, background nonuniformity and other artifacts, identifying these points is not straightforward in real images.

**Profile denoising:** before analysis, every profile is denoised by local polynomial approximation on adaptive neighborhoods ([12]), which preserves coarse-scale discontinuities. This algorithm requires to be tuned on the noise variance, estimated using [5].

**Initialization of background and ball intensity:** a rough estimate f' for the ball intensity is obtained by analyzing the histogram of values along the profile p. If the ball is expected to be lighter than the background, as  $c_1$  and  $c_2$  overlap significantly, the highest peak in the second half of the histogram is taken. Whenever the ball is darker than the background, we simply invert the image before processing. Estimation of ball color is performed separately on each profile, in order to increase the robustness to lighting variations. For the same reason, two background colors are considered; they are initialized with the average of a fixed number N of pixels at the beginning and at the end of the

profile: 
$$b'_1 = \sum_{x=0}^{N} \frac{p(x)}{N}$$
 and  $b'_2 = \sum_{x=E-N}^{E} \frac{p(x)}{N}$ .

**Initialization of interest points** given the index  $\tilde{F}$  along the profile of a pixel having the ball intensity f', an initial estimation  $B'_1$ ,  $B'_2$  of  $B_1$ ,  $B_2$  is given as:

$$B'_{1} = \max_{B} \{ B < \tilde{F}, \, p(B) < b_{1} + M_{1} \widehat{\sigma} \}$$
 (3)

$$B'_{2} = \min_{B} \{ B > \tilde{F}, \, p(B) < b_{2} + M_{1} \hat{\sigma} \}, \qquad (4)$$

where  $M_1$  is a tuning parameter, and  $\hat{\sigma}$  is the noise variance previously estimated. Refined background colors  $b_1$  and  $b_2$ are obtained by averaging values on  $[O, B'_1]$  and  $[B'_2, E]$ . A

<sup>&</sup>lt;sup>2</sup>The ball is assumed to be sufficiently far from the vanishing point, and the pixel intensity is assumed proportional to the time it has been covered by the ball.



## Figure 3. Intensity profiles along directions approximately parallel to the blur direction in the image have similar characteristics.

similar procedure is implemented for estimating  $F'_1$  and  $F'_2$  as well as to refine f' to f.

Line fitting and final estimate a straight line  $l_1$  ( $l_2$ ) is fitted to  $p_i$  values on a central part of interval  $[B'_1, F'_1]$  $([F'_2, B'_2])$ . The width of this interval is given as a tuning parameter. Line fitting is performed by the iteratively reweighted least squares algorithm given by matlab robustfit command. The intersection of lines  $l_1$  and  $l_2$ with the intensities  $b_1$ ,  $b_2$  and f gives  $B_1$ ,  $B_2$ ,  $F_1$ ,  $F_2$ . The last two steps are iterated once for further refinement.

The profile analysis procedure is sufficiently robust to small errors in the estimation of  $\hat{\theta}$ , as well as mild ball shading and background nonuniformity.

# 3.3 Reconstructing the ball position and speed

Once the points  $\{B_1^i, F_2^i\}_{i=1,\dots,n}$  ( $\{F_1^i, B_2^i\}_{i=1,\dots,n}$ ), belonging to  $c_1$  ( $c_2$ ) are extracted from the *n* intensity profiles, the ellipse  $c_1$  ( $c_2$ ) is estimated by conic fitting.

The procedure is simplified by some geometric constraints on  $c_1$  and  $c_2$ . In particular, given the perspective camera calibration parameters, we can constrain  $c_1$  ( $c_2$ ) to be the image of a sphere; this fixes 2 of the 5 degrees of freedom of a conic. On the contrary although the rectilinear bounds of the smear are usually visible, they are not used to further constrain the solution (by requiring that  $c_1$  and  $c_2$  are tangent to both), as theoretic and experimental evidence ([4]) shows that ordinary edge extraction techniques are affected by systematic error in locating these contours.

Once  $c_1$  and  $c_2$  (the image of the ball at the first and second curtain) is known, the ball position at the beginning and end of the exposure can be easily reconstructed by means of basic projective geometry, provided that the sphere radius is known and the camera is calibrated. The vector connecting their centers is the 3D displacement occured during the exposure. This also allows us to compute the absolute speed of the ball whenever the exact exposure time  $\Delta t$  is known (which is often the case).

## 4 Experimental results

The proposed procedure has been validated both with synthetic and camera images. Synthetic tests, in particular, have been set up to compare results to ground truth, which is hardly available on real data.

The synthetic dataset was composed by two images having different amounts of perspective effect (see Figure 4, first two rows). These have been rendered using the popular 3D modeling software *Blender*, creating an 800x600 pixel blurred image with 90° horizontal field of view, by averaging 100 frames of an animation, according to (1). Performance metrics are computed exploiting the camera calibration matrix and ground truth: we consider the average of the sphere center localization errors at the first and second curtain. Tests have been run by taking into account several resolutions and noise levels. Table 1 reports the localization error averaged over the two images and 10 different realizations of noise on each.

The real dataset has been collected by using a 2MPixel Canon A60 camera, photographing white and colored table tennis balls, and other spheres of different diameter and material such as a mouse ball. No special attention was devoted to lighting, therefore the "uniform-intensity ball image" assumption is not exactly met: while our technique did not work under severe shading or with shiny materials such as metal, most other scenarios were handled correctly. In general, quasi-lambertian materials (such as rubber covering the mouse ball) have been easily handled. Depending on the ball diameter and speed, we used exposure times ranging from 1/200 to 1/25 seconds. Often, for significant ball speeds, that was the fastest acceptable shutter speed for achieving sufficient exposure, even using the widest lens aperture stop f/2. All camera images have been shot using the widest available field of view, to maximize perspective effects.

	Image res	Ball image width (px)	Mean error	AWN $\sigma$	Average error
100%	800x600	$170 \div 130$	$0.012 \cdot R$	.000	$0.001 \cdot R$
75%	600x450	$127 \div 97$	$0.021 \cdot R$	.005	$0.012 \cdot R$
50%	400x300	$85 \div 65$	$0.110 \cdot R$	.010	$0.040 \cdot R$
25%	200x150	$42 \div 32$	$0.600 \cdot R$	.015	$0.211 \cdot R$

Table 1. Top: mean reconstruction error (fraction of the ball radius R) w.r.t. image resolution; AWN  $\sigma = .005$ . Bottom: mean reconstruction error (fraction of the ball radius R) w.r.t. noise variance (full resolution).



Figure 4. First and second rows: relevant part of a synthetic 800x600 image with AWN ( $\sigma = .005$  of the image dynamic range); fitted ellipses for  $c_1$  and  $c_2$  (blue), and ground truth (red); reconstructed position and velocity with different noise realizations (blue), and ground truth (red). Bottom row: real images and estimated  $c_1$  and  $c_2$ .

Most cameras apply a logarithm-like transfer function to the pixel intensity values returned by their sensors, which partially invalidates our analysis since we assume a linear transfer function. This manipulation of the sensor data can be avoided by using digital cameras with RAW shooting mode, or by compensating it applying the inverse camera response function (see [9]). This issue is common to many systems exploiting radiometry.

Our technique does not handle saturation (clipping) of intensity values, which happens whenever the ball is overexposed: in this case, in fact, the intensity value of a pixel is not directly related to the coverage time by the ball image. Saturation can be avoided by choosing appropriate exposure times and aperture.

In the current unoptimized Matlab implementation, processing a typical image requires several seconds. The computational effort is dominated by the profile analysis step: considering fewer profiles results in significant speedup, while degrading the localization accuracy. Since there are no intrinsically expensive steps in our procedure, we expect that an optimized implementation could work in realtime on video frames.

## 5 Conclusions and future works

We provided a method for reconstructing the position and velocity of a moving ball from a single perspective calibrated image, in which the ball image appears blurred. The algorithm analyzes several 1D image intensity profiles in order to extract points belonging to the ball contours at the beginning and the endo of the exposure; then, a geometric procedure allows to reconstruct the ball location and displacement during the exposure, provided that the camera calibration parameters are known.

The technique has been successfully tested both on synthetic and camera images, obtaining convincing results even on camera images where our assumptions are not exactly met because of shading and other variations in the ball and background intensities.

An optimized implementation is underway, for analyzing and augmenting sport videos in realtime. We are also currently working on integrating a preprocessing step, to expand the applicability scenario of the algorithm to severely nonuniform backgrounds or balls with nonuniform pigmentation such as a soccer ball. Therefore, besides the global displacement vector, the ball spin (both axis and speed) can be estimated by locally analyzing the blur of features on the surface, exploiting [3, 2].

## References

[1] B. Bascle, A. Blake, and A. Zisserman. Motion deblurring and super-resolution from an image sequence. In *proc. of*  ECCV 1996 Conference, pages 573-582, 1996.

- [2] G. Boracchi and V. Caglioti. Corner displacement from motion blur. In proc. of ICIAP 2007 Conference, 2007.
- [3] G. Boracchi and V. Caglioti. Motion blur estimation at corners. In proc. of VISAPP 2007 Conference, 2007.
- [4] V. Caglioti and A. Giusti. Ball trajectory reconstruction from a single long-exposure perspective image. In *Proc. of CVBASE 2006 Workshop*, 2006.
- [5] D. L. Donoho and I. M. Johnstone. Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, 81(3):425–455, 1994.
- [6] R. Fergus, B. Singh, A. Hertzmann, S. T. Roweis, and W. Freeman. Removing camera shake from a single photograph. ACM Transactions on Graphics, Proc. of SIGGRAPH 2006 Conference, 25:787–794, 2006.
- [7] W. T. Freeman and E. H. Adelson. The design and use of steerable filters. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 13(9):891–906, 1991.
- [8] A. O. Gopal Pingali and Y. Jean. Ball tracking and virtual replays for innovative tennis broadcasts. In *proc. of ICPR* 2000 Conference, page 4152, Washington, DC, USA, 2000. IEEE Computer Society.
- [9] M. D. Grossberg and S. K. Nayar. Modeling the space of camera response functions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(10):1272–1282, 2004.
- [10] G. J. J Ren, J. Orwell and M. Xu. A general framework for 3d soccer ball estimation and tracking. In *Proc. of ICIP 2004 Conference*, 2004.
- [11] L. C. Jonathan Rubin, Burkhard C. Wuensche and C. Stevens. Computer vision for low cost 3-d golf ball and club tracking. In *Proc. of Image and Vision Computing New Zealand*, 2005.
- [12] V. Katkovnik, A. Foi, K. Egiazarian, and J. Astola. Directional varying scale approximations for anisotropic signal processing. In *proc. of EUSIPCO 2004 Conference*, 2004.
- [13] T. Kim, Y. Seo, and K.-S. Hong. Physics-based 3d position analysis of a soccer ball from monocular image sequences. In proc of ICCV 1998 Conference, pages 721–726, 1998.
- [14] H.-Y. Lin. Vehicle speed detection and identification from a single motion blurred image. In proc. of WACV 2005 Workshop, 2005.
- [15] H.-Y. Lin and C.-H. Chang. Automatic speed measurements of spherical objects using an off-the-shelf digital camera. In proc. of ICM 2005 Conference, 2005.
- [16] Y. Ohno, J. Miura, and Y. Shirai. Tracking players and estimation of the 3d position of a ball in soccer games. In proc. *ICPR 2000 Conference*, pages 1145–1148, 2000.
- [17] I. D. Reid and A. North. 3d trajectories from a single viewpoint using shadows. In proc. of BMVC 1998 Conference, 1998.
- [18] Y. Yitzhaky and N. S. Kopeika. Identification of blur parameters from motion-blurred images. In *Proc. SPIE Vol. 2847*, pages 270–280, Nov. 1996.