Novel Patterns and Methods for Zooming Camera Calibration

Andrea Pennisi, Domenico Bloisi, Claudio Gaz, Luca Iocchi, Daniele Nardi
Department of Computer, Control, and Management Engineering
Sapienza University of Rome
via Ariosto 25
00185, Rome, Italy
{pennisi,bloisi,gaz,iocchi,nardi}@dis.uniroma1.it

ABSTRACT

Camera calibration is a necessary step in order to develop applications that need to establish a relationship between image pixels and real world points. The goal of camera calibration is to estimate the extrinsic and intrinsic camera parameters. Usually, for non-zooming cameras, the calibration is carried out by using a grid pattern of known dimensions (e.g., a chessboard). However, for cameras with zoom functions, the use of a grid pattern only is not sufficient, because the calibration has to be effective at multiple zoom levels and some features (e.g., corners) could not be detectable. In this paper, a calibration method based on two novel calibration patterns, specifically designed for zooming cameras, is presented. The first pattern, called in-lab pattern, is designed for intrinsic parameter recovery, while the second one, called on-field pattern, is conceived for extrinsic parameter estimation. As an application example, on-line virtual advertising in sport events, where the objective is to insert virtual advertising images into live or pre-recorded television shows, is considered. A quantitative experimental evaluation shows an increase of performance with respect to the use of standard calibration routines considering both re-projection accuracy and calibration time.

Keywords

Virtual advertisement, zooming camera calibration, augmented reality.

1 INTRODUCTION

Having an accurate calibration is crucial in many computer vision applications, ranging from automatic video surveillance to augmented reality. In order to calibrate a non-zooming camera, a grid pattern with known dimensions can be used. Usually a chessboard with black and white squares is chosen, allowing to obtain a good correspondence between image and world points [Har04a].

However, when the camera has the capability of zooming, the use of a chessboard only is not sufficient. Indeed, since the calibration has to be carried out at different zoom levels, the accuracy of the correspondence between scene and image points can decrease, especially at high zoom levels.

In particular, when shooting sport events a lot of changes in the zoom levels are required to capture objects that can be at different distances from the camera. Using standard multi-level calibration techniques (e.g., [AlA06a, May97a, Stu97a]) can introduce re-projecting errors. Indeed, since the calibration is repeated at different zoom levels, each calibration is influenced by the error introduced at the previous zoom level, thus inevitably increasing the final re-projecting error.

In this paper a calibration method based on two novel calibration patterns, specifically designed for variable-zoom cameras, is presented. The two patterns allow to calculate a set of calibration parameters that do not depend on the camera zoom level. The first pattern is designed for computing an accurate calculation of the camera intrinsic parameters even at high zoom levels, while the second one is conceived for reducing the time needed to calculate the extrinsic parameters of the camera and for making easier and faster the on-field calibration process.

The method has been applied to a real system for adding virtual advertisement in live sport events that is currently used by Duel TV S.p.A.[1], a company that provides real time virtual advertisement services for sport events. In order to quantitatively evaluate the performance of our calibration approach, it is compared with

[1] www.dueltv.com

the previous one used by the company, taking into account both the precision in placing the virtual billboards and the calibration time.

The remainder of the paper is organized as follows. After the analysis of related work in Section 2, Section 3 describes the two novel patterns and the calibration routines to estimate the camera parameters. A real-world application scenario is described in Section 4, while Section 5 shows experimental results. Conclusions are drawn in Section 6.

2 RELATED WORK

Several solutions have been proposed in literature to find methods that aim at achieving an accurate camera calibration (see [Sal02a] for a detailed survey). The use of zooming cameras adds a further level of complexity, since a high accuracy may be maintained over different levels of zoom. A set of calibration methods for zooming cameras are discussed in the following.

Al-Ajlouni and Fraser in [AlA06a] propose a zoom-dependent calibration process, whereby the traditional image coordinate correction model for camera interior orientation and lens distortion is expressed as a function of the nominal zoom focal length written into the EXIF header of the image file. This removes the requirement of using fixed zoom/focus settings for the images forming the photogrammetric network. However, the authors assume that the computation of the empirically determined Z-D calibration parameters is unlikely to be required too frequently, that is a too strong assumption for real-world settings (e.g., sport events).

Maybank and Faugeras in [May97a] estimated the intrinsic parameters of the camera considering that some constraints based on the epipolar geometry of two views are related to the rigidity of camera movements. To compute the parameters of the epipolar equation, they select some points in the current scene and track them in the image while moving the camera with a random motion. The main drawback of this method is the loss of accuracy generated by the manual selection of the points.

Strum in [Stu97a] proposes to set unknown parameters in order to solve the problems in [May97a]. An interdependence model of intrinsic parameters is computed in a pre-calibration phase. In such a way, the calibration is reduced to the estimation of only one parameter from which all the others depend. However, it is not clear if this method is effective for imaging systems where the interdependence of parameters follows a more complicated model with respect to the one described by the author.

Hyunwoo and Ki Sang in [Kim00a] propose a method for simplifying the computation of the camera intrinsic parameters. The authors are able to overcome the degenerate configurations and to get a closed form solution. They introduce also a non-linear algorithm that adjusts both camera parameters and inter-image homography, thus a more accurate image registration becomes possible. However, even if such a method is useful for moderate camera rotations and zooming variations, there is an increase of failures for wider camera movements.

Oh *et al.* in [Oh09a] present a method to calibrate pan-tilt-zoom-focus cameras using both a pattern- and a rotation-based calibration approach. The method is composed by two separate procedures for calibrating zoom and focus respectively. The zoom calibration is based on the detection of a set of known patterns (chessboards) and it is carried out at different zoom levels. The focus calibration is obtained considering first the lowest focus value of the camera, and then, for the remaining focus values, by applying an automatic procedure, that takes into account the previous calculated zoom calibrations. This approach has good results for real cameras with translation offsets, but it generates incorrect results at high-zoom levels. Moreover, it uses 10 calibration patterns and the calibration must be repeated for all zoom levels.

Sinha *et al.* in [Sin06a] deal with the problem of estimating the parameters of the calibration model for active pan-tilt-zoom cameras. The camera intrinsic parameters are estimated over its full range of pan, tilt, and zoom by computing homographies between images acquired by a rotating and zooming camera. The calibration algorithm also computes accurate calibrated panoramas at multiple levels of detail. The main weakness of this method is the need of re-estimating the calibration parameters every time the camera moves.

Sarkis *et al.* in [Sarkis07a] propose a least-square approach to model the variation of internal parameters as a function of focus and zoom. This approach is able to increase the accuracy in modelling the intrinsic parameters of a zoom lens camera system and to decrease the pixel re-projection error. However, the method is sensitive only to large variations of focus and zoom values.

Finally, Agapito *et al.* in [Aga01a] use an homography to compute intrinsic parameters, assuming that the aspect-ratio and the principal point are fixed in time, while the focal length changes as the camera moves. They assume also that the principal point is the best of the aspect ratio and it can be considered as the image center. Given the principal point, it becomes simpler to auto-calibrate the camera. The main drawback of this method is the need to find the principal point, that is a non-trivial task.

All of the above discussed methods have limitations that make them not robust to zoom variations and/or not easy to set up. In the next sections we will describe

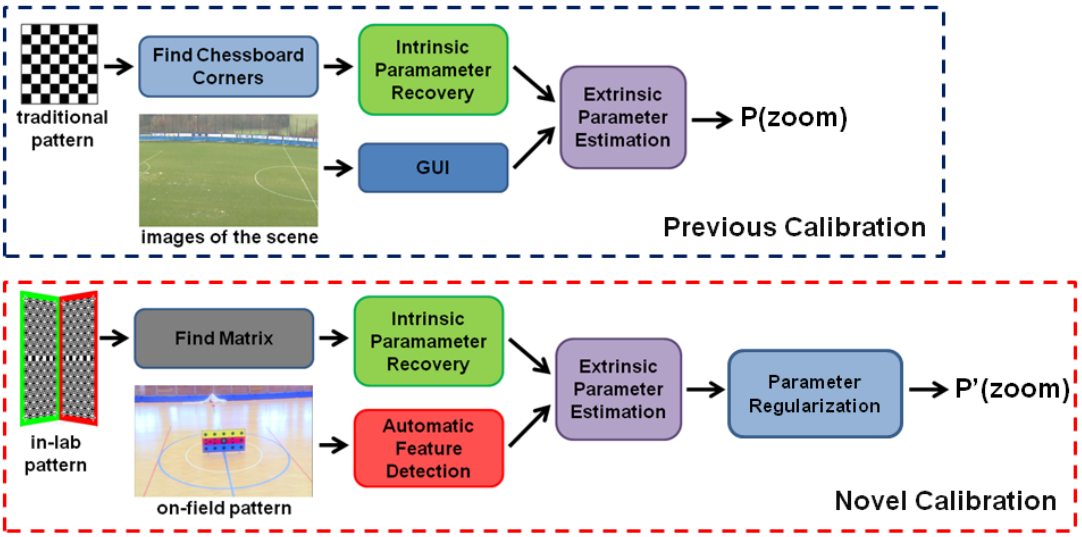

Figure 1: The proposed approach. With respect to common calibration methods (in this figure the one previously used by the company is reported), in the novel method two different calibration patterns are used for in-lab and on-field calibration, respectively.

our approach that aims at providing an easier calibration procedure that is robust to zoom variations.

3 OVERALL APPROACH

In order to achieve the two goals of camera calibration (namely, intrinsic parameter recovery and extrinsic parameter estimation) [Har04a], we propose the use of two novel calibration patterns. With respect to common calibration methods, the proposed patterns allow to obtain a more accurate intrinsic parameter calculation and an automatic extrinsic parameter estimation, with the possibility of regularizing camera parameters in function of the zoom levels (see Fig. 1).

The first pattern, called *in-lab pattern*, is used to recover the intrinsic camera parameters. The *Find Matrix* module is responsible for detecting the features in the pattern that are sent as input to the *Intrinsic Parameter Recovery* module. Such a module uses standard routines [Sir04a] to calculate the intrinsic parameters of the camera. A second pattern, called *on-field pattern*, allows to estimate the extrinsic parameters in a fully automatic way thanks to the *Automatic Feature Detection* module. The introduction of the on-field pattern replaces the graphical user interface (GUI) that was used in the previous calibration method to associate image points in the captured frame to known world coordinates (e.g., soccer field line). A novelty is also represented by the *Parameter Regularization* module, that aims at refining the calibration parameters at different zoom levels.

3.1 In-lab Pattern

To achieve an accurate calibration even in presence of high zoom levels, we focussed our analysis on the calibration pattern. First, we analysed the results obtained using a traditional chessboard (with black and white squares) at different zoom levels. To compute the results, we used the methods provided by the OpenCV library (version 2.4.4) [Ope13a].

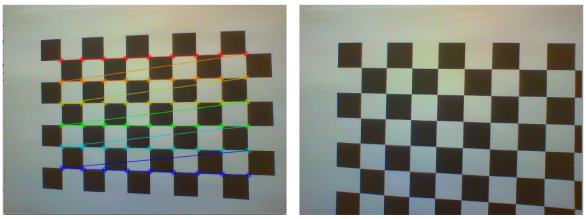

Figure 2: Traditional calibration using a chessboard pattern. a) The OpenCV routines can detect all the chessboard corners. b) The corners cannot be associated to the corresponding points on the chessboard if the pattern is not completely captured.

We observed that if the chessboard is entirely captured, then the calibration accuracy is good (Fig. 2a). However, for high zoom levels, it can happen that not all the corners are captured and, as a consequence, it is impossible to carry out the calibration, because the found corners cannot be associated with the corresponding ones on the chessboard pattern (Fig. 2b). Therefore, a pattern allowing for a non-ambiguous association even if it is not completely captured should be used.

It is worth noting that, in order to calibrate a zooming camera, the focus parameter has to be maintained stable and, for this reason, increasing the zoom level can generate blurred images. Since the squares are effective features to be detected at medium and low zoom levels, an additional geometric figure to be used in combination with squares must be individuated. Such geometric

figure has to be detectable even in presence of blurred images.

Taking into account the above considerations, we developed a novel calibration pattern that is shown in Fig. 3a. It is a 11×11 chessboard containing 454 circles inside. Since the pattern is made of two different types of features (i.e., squares and circles), it is possible to consider such features together or separately, depending on the zoom level, to calibrate the camera.

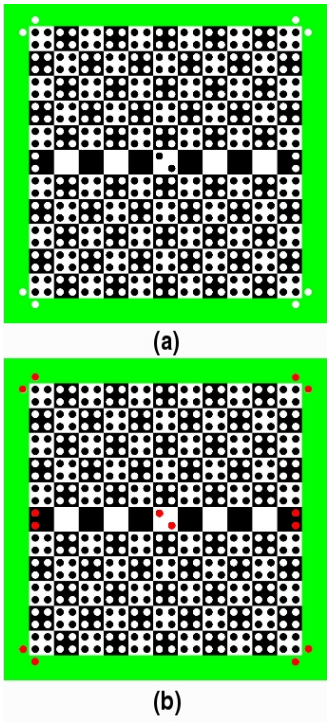

Figure 3: a) The novel calibration pattern. b) The 14 reference points highlighted in red.

The pattern has a coloured (green) border that is useful for individuating it in the scene when low zoom levels are used. There are also 14 reference circles that can help in understanding which region of the pattern is currently viewed by the camera, in case the pattern is only partially captured (Fig. 3b).

3.2 Find Matrix

The Find Matrix module is responsible to detect the squares and the circles in the image and to find the correspondences between the found features and the model pattern (shown in Fig. 3a). The circle centres are individuated using an algorithm (see Alg. 1) based on the following OpenCV [Ope13a] functions: *threshold*, *findContours*, *approxPolyDP*, and *minEnclosingCircle*.

Given the captured frame, a binary image *BImg* is obtained applying a thresholding process, then the above listed functions are used to find the contours in *BImg*, the approximation of the contours to the best fitting

Input:
binary image BImg;
Output:
set of pairs $(center, radius)$ $< ctrs, r >$;
Data Structures:
set of contours cont;
set of closed coutours contPol;
Initialize:
$\forall_i \; ctrs[i] = 0, cont[i] = 0, contPol[i] = 0, r[i] = 0$;
$< cont > \leftarrow findContours(BImg)$;
for *cont* $[i]$ **do**
$contPol \leftarrow approxPolyDP(cont, \varepsilon, closed)$;
$< ctrs, r > \leftarrow minEnclosingCircle(contPol)$;
$< ctrs, r > \leftarrow RefineCenters(< ctrs, r >)$;

Algorithm 1: Find Circle Centers

polygons, and the approximation of those polygons to circles. Notice that the pattern does not need to be perfectly aligned or rectified in the image, since corner and circle detection are robust enough to variations of orientation of the pattern.

In order to eliminate the false positive detections, the list of centres is refined by eliminating the polygons that are too large or too small according to a parametric threshold T, computed as:

$$T = \frac{\sum_{i=1}^{n} radius_i}{n} \tag{1}$$

where $radius_i$ is the i-th radius belonging to the set S of detected circles and n is the cardinality of S.

We observed that when the zoom level is low, i.e., the captured image of the pattern is small with respect to the camera frame, many false positives arises in detecting the circles due to the difficulty of extracting well defined shapes. However, for high zoom levels, even if the image of the pattern can result blurred, our algorithm is able to find all the circles in the visible portion of the pattern as well as all the circle centres (Fig. 4).

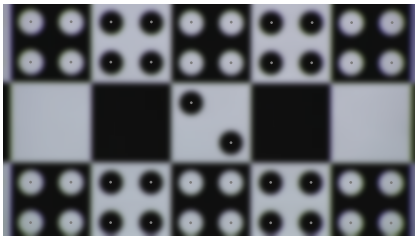

Figure 4: The *Find Circle Centres* algorithm is robust even in presence of a blurred image as input.

For doing this, the *Find Matrix* Algorithm (Alg. 2) takes as input the dimensions w and h of the circle grid in the model pattern, a point p (namely, the point that has the smaller abscissa among all the found points in the pattern), four threshold values T, $offset$, δ, ε

Input: $< ctrs, r >,\ T,\ offset,\ p$
Output: M
Local Variables:
$thresholds\ \delta,\ \varepsilon$
Data Structures:
$set\ of\ current\ row\ points\ rpnts$
$binary\ matrix\ M$
$average\ of\ current\ set\ of\ radii\ av$
Inizialize:
$\forall_{i,j}\ M(i,j) = 0, \forall_k\ rpnts[k] = 0$

for each $ctrs[i]$ **do**
  $< rpnts > \leftarrow findRow(ctrs, i)$
  $av \leftarrow computeAverage(rpnts)$
  for each $rpnts[j]$ **do**
    $\delta \leftarrow av + p + (j * T) - offset$
    $\varepsilon \leftarrow av + p + (j * T) + offset$
    if $rpnts[j] < \varepsilon, rpnts[j] > \delta$ **then**
      $M(l,m) = 1;$
    else
      $M(l,m) = 0$
    $deleteCenters(rpnts, < ctrs, r >)$
$M \leftarrow CropMatrix(M)$

Algorithm 2: Find Matrix

that are related to the dimensions of the pattern and can be experimentally found, and a set of pairs $< ctrs, r >$ (i.e., the list of the found centres together with their relative radii).

The procedure starts by initialising with zeros the $w \times h$ matrix M. Then, the algorithm takes the first element in the list $ctrs$ and searches for points that have the ordinate value included in a range defined by the thresholds δ and ε. All the points that satisfy such constrains form a *circle row*. The obtained row is sorted according to the abscissa values of its points and it is compared to the corresponding row model of the known pattern. A vector $rpnts$ is filled with values "1" if the point is included in the row model, "0" otherwise. By iterating the above steps, a matrix M composed by "1" and "0" is obtained. Finally, if M is smaller than the model pattern, a cropping procedure is applied to reduce its dimensions.

Once M has been computed, a graph matching between M and a matrix C, that represents the model pattern, can be applied. It is worth noting that each image used for the calibration procedure may contain at least a reference point in it. This is a reasonable assumption, since the reference points are distributed along the whole pattern (see Fig. 3b). Thus, we assume that the captured portion of the pattern is composed by a set of circles and by at least one group of reference points. Since the patterns of the reference points are all different each other, the observation of a single one is sufficient to determine the portion of the calibration pattern that is captured.

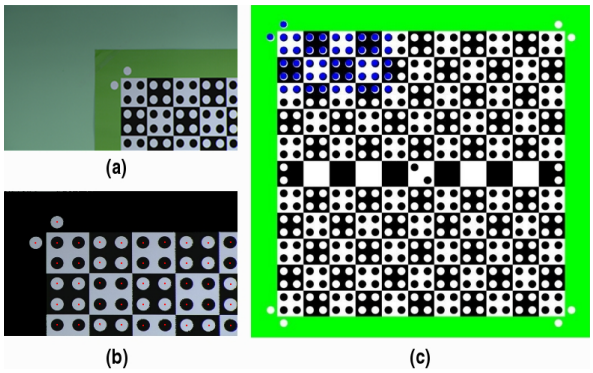

Figure 5: Recognition example for a partially captured pattern. a) The image in input. b) The detected centers. c) The graph matching procedure correctly assigns the observed centres to the corresponding ones in the model pattern.

As an example, if the current image of the pattern is the one shown in Fig. 5a, the matrix M will be:

$$M = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

Given the matrix C that represents the model pattern:

$$C = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 & 0 & \cdots & 0 & 1 & 0 \\ 1 & 1 & 1 & \cdots & 1 & 1 & \cdots & 1 & 1 & 1 \\ 0 & 1 & 1 & \cdots & 1 & 1 & \cdots & 1 & 1 & 0 \\ 0 & 1 & 1 & \cdots & 1 & 1 & \cdots & 1 & 1 & 0 \\ \vdots & \vdots & \vdots & \cdots & \vdots & \vdots & \cdots & \vdots & \vdots & \vdots \\ 0 & 1 & 1 & \cdots & 1 & 1 & \cdots & 1 & 1 & 0 \\ 0 & 1 & 0 & \cdots & 1 & 0 & \cdots & 0 & 1 & 0 \\ 0 & 1 & 0 & \cdots & 0 & 1 & \cdots & 0 & 1 & 0 \\ 0 & 1 & 1 & \cdots & 1 & 1 & \cdots & 1 & 1 & 0 \\ \vdots & \vdots & \vdots & \cdots & \vdots & \vdots & \cdots & \vdots & \vdots & \vdots \\ 0 & 1 & 1 & \cdots & 1 & 1 & \cdots & 1 & 1 & 0 \\ 0 & 1 & 1 & \cdots & 1 & 1 & \cdots & 1 & 1 & 0 \\ 1 & 1 & 1 & \cdots & 1 & 1 & \cdots & 1 & 1 & 1 \\ 0 & 1 & 0 & \cdots & 0 & 0 & \cdots & 0 & 1 & 0 \end{bmatrix}$$

it is possible to find the portion of C corresponding to M (see Fig. 5c).

Fig. 6 shows the in-lab pattern that we used. It is the combination of two above discussed model patterns (one with red and one with green borders). The dimensions for each one of the two parts of the in-lab pattern are 2 m × 1 m with 18 rows and 34 columns of circles, each circle having a radius of 1.5 cm and being at a distance of 5 cm (center-center) from the others. The two parts of the in-lab pattern are positioned with an angle of 90 degrees in order to have non-coplanar points. The two different border colours facilitate the detection of the in-lab pattern in the scene.

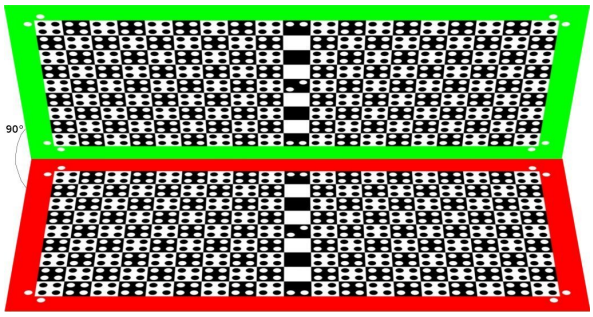

Figure 6: In-lab pattern.

3.3 Intrinsic Parameter Recovery

A series of images containing the in-lab pattern at different zoom levels are collected to determine the intrinsic parameters. Once the key points are obtained using the Find Matrix module, a common technique [Sir04a] is applied to calculate five intrinsic parameters: f: focal length; (s_x, s_y): pixel size in x, y; (u, v): principal point. Usually, it is possible to assume square pixels and so $s_x = s_y = s$. The intrinsic matrix K can be defined as follows:

$$K = \begin{bmatrix} f_x & s & u_0 \\ 1 & f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix}$$

where s is the skew parameter that represents the angle between the x and y pixel axes, $f_x = f/s_x$, $f_y = f/s_y$ are the focal lengths.

The quantitative evaluation for the in-lab calibration will be given in Section 5.

3.4 On-field Pattern

The on-field step of the proposed method requires the use of a second calibration pattern (Fig. 7a), made of three coloured rectangles as background, 19 black circles, and a white "+" sign drawn in the central black circle (Fig. 7b). We realized an on-field pattern of 2 m width and 1 m height.

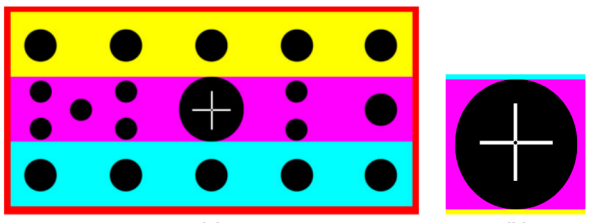

Figure 7: a) On-field pattern. b) A detail of the "+" sign in the center of the pattern.

In order to perform the extrinsic parameter estimation of the camera, two short videos have to be collected on the field:

1. The pattern is captured from five different angles of views with a fixed zoom level (Fig. 8);
2. The center of the pattern is captured starting from the maximum zoom level until reaching the minimum level (Fig. 9).

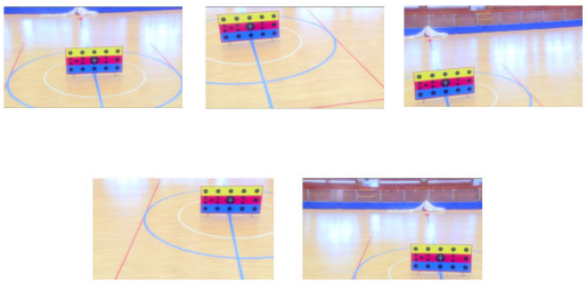

Figure 8: The on-field pattern is captured from 5 different angles of view.

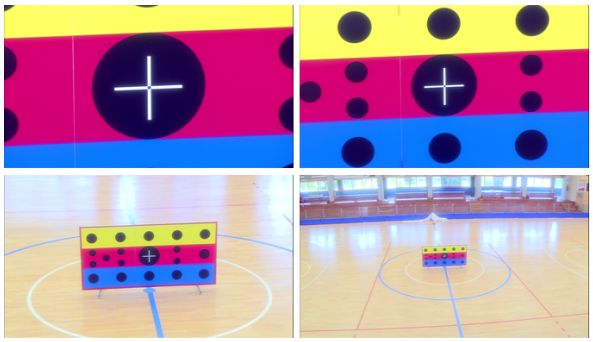

Figure 9: The "+" sign in the center of the on-field pattern is captured at different levels of zoom.

Each video is about 2 minutes long. The aim of the first video is to obtain a set of image points positioned in the center and in the four corners of the image. The second video is used to track the "+" sign over different levels of zoom. The extrinsic parameter calculation using the two videos is detailed in the following.

3.5 Extrinsic Parameter Estimation

Extrinsic parameters play a crucial role in the camera calibration process, since those parameters define the location and orientation of the camera with respect to the world reference frame.

To obtain such parameters the algorithm proposed by Tsai in [Tsa86a] and the calibration method by Davis and Chen [Dav03a] are used.

Let (u, v) be the ideal (distortion-free) pixel image coordinates, and $(\tilde{u}, \tilde{v})$ the corresponding real observed image coordinates. The ideal points are the projection of the model points according to the pinhole model. Similarly, (x, y) and $(\tilde{x}, \tilde{y})$ are the ideal (distortion-free) and real (distorted) normalized image coordinates. As reported in [Zhang00a] we have:

$$\tilde{x} = x + x[k_1(x^2+y^2) + k_2(x^2+y^2)^2]$$
$$\tilde{y} = y + y[k_1(x^2+y^2) + k_2(x^2+y^2)^2]$$

where k_1 and k_2 are the coefficients of the radial distortion.

The function for the radial distortion is here approximated through the first two terms of the development of the series $f(r) = 1 + k_1 r^2 + k_2 r^4 + k_3 r^6 + ...$, in such a way that only the parameters k_1 and k_2 are considered.

3.6 Parameter Regularization

In order to regularize the intrinsic and extrinsic parameter functions (depending on zoom levels), several methods can be applied. Since no information about the physics of the optical phenomena have been used, a "black box" approach is suitable. We compared polynomial fitting versus Artificial Neural Network (ANN) [Fan13a] and chose the latter because it provides better results.

The second video collected during the on-field calibration phase, that contains the "+" sign captured at different zoom levels, is used to refine the calibration parameters, in particular, the principal point (u, v) and the focal lengths f_x and f_y, and the radial distortion coefficients k_1 and k_2 are considered.

Two different ANNs have been implemented, the former for managing the lower zoom levels, the latter for the higher ones. Indeed, from Fig. 10 it is noticeable that the trends for f_y (Fig. 10a) and for the first coefficient of radial distortion k_1 (Fig. 10b) are quasi-linear for low zoom levels, while they become non-linear for higher zoom levels.

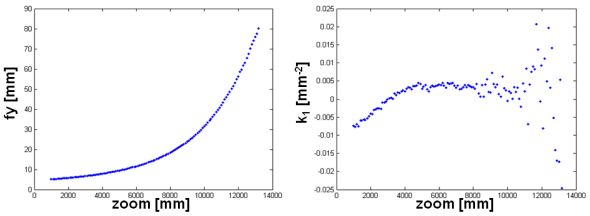

Figure 10: Variation of camera parameters in function of zoom. a) Focal length along y axis. b) First coefficient of radial distortion.

The neural network for the lower zoom levels includes a hidden layer made of two units, while the neural network for the higher levels has a hidden layer composed by six units. This choice has been taken for allowing the ANN that manages the non-linear zone to have a greater number of degrees of freedom, while inserting more units in the hidden layer of the ANN that manages the quasi-linear zone would have generated overfitting problems. Two results of the regularization process are shown in Fig. 11.

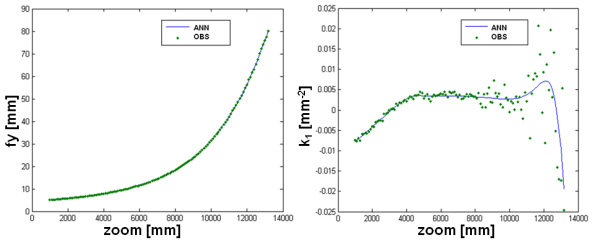

Figure 11: Real observations (OBS) are reported as green dots, while the functions find by the two ANNs are reported as blue lines. a) Focal length along y axis. b) First coefficient of radial distortion.

In order to allow a smooth transition between the two regularization functions calculated by the ANNs, a third (overlapping) zone, where a regularization function is computed as a linear combination of the other two functions, is considered (see Fig. 12). Let f_1 be the return value of the first neural network computed on a value x and f_2 be the function generated by the second neural network for the same value x. Let I_0 and I_1 be the two extremities of the central green stripe in Fig. 12, we have:

$$\lambda = \frac{x - I_0}{I_1 - I_0};$$

and $f_3(x)$ will be:

$$f_3(x) = (1-\lambda) f_1(x) + \lambda f_2(x)$$

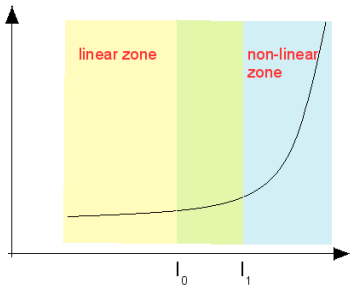

Figure 12: Overlapping zone where a regularization function is computed as a linear combination of the two functions computed by the ANNs.

4 APPLICATION SCENARIO

Virtual advertising concerns the use of computer vision techniques to insert virtual advertising images into live or pre-recorded television shows (e.g., sport events). The aim is to put virtual advertising billboards[2] on the play field (Fig. 13).

[2] The trade marks shown in Figs. 13 and 14 have been randomly chosen for demonstrating the output of the system and are not involved in the work described in this paper.

Figure 13: Virtual advertising example. Virtual billboards (highlighted by the red ellipses) are positioned in the scene using a chroma-keying technique.

This particular application scenario is a challenging one, since a lot of changes in the zoom levels are required for adequately capturing sport events. Thus, an accurate camera calibration is needed for the replacing process, since the projection of the billboard to be replaced onto the image plane must be known. Moreover, the calibration model must be robust to camera zoom variations, since the replacement must be dynamic to adapt to the varying zooming parameters (Fig. 14).

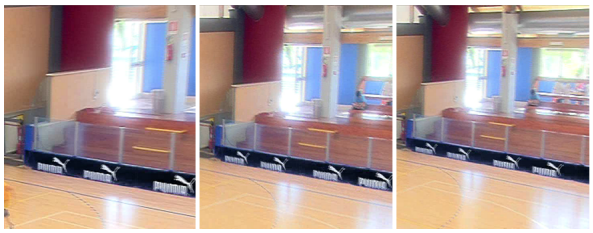

Figure 14: The proposed approach is robust to zoom variations. This figure shows three different screen shots resulting from three different zoom values.

5 EXPERIMENTAL RESULTS

A quantitative evaluation has been carried to demonstrate the effectiveness of the approach. For measuring the accuracy of the in-lab calibration, a set of views of the in-lab pattern captured with a professional HD camera (1920×1080 resolution) has been collected. The accuracy of the intrinsic parameters has been calculated at different zoom levels and with different angles of view.

The results generated by using our novel calibration pattern are compared with those obtained by using a traditional pattern. The comparison has been performed by manually determining a ground truth that has been compared with the re-projection values generated by the two methods. The quantitative results of the evaluation are reported in Table 1 and the error is measured in terms of pixels. For each considered zoom level 20 different images have been examined.

The error values are also plotted in Fig. 15 to highlight that the proposed method generates a constant error (about 0.5 pixel), while the traditional procedure introduces an higher error with a more variable trend.

Zoom Level	Trad. Calib. Error		Prop. Calib. Error	
	Avg.	Std. Dev.	Avg.	Std. Dev.
1000	0.68	0.32	**0.56**	0.26
1500	0.70	0.34	**0.52**	0.25
1800	0.75	0.36	**0.53**	0.25
2000	0.70	0.33	**0.51**	0.24
2400	0.72	0.35	**0.53**	0.26

Table 1: Quantitative comparison of the re-projection error obtained by using a traditional pattern based calibration and by adopting the proposed method.

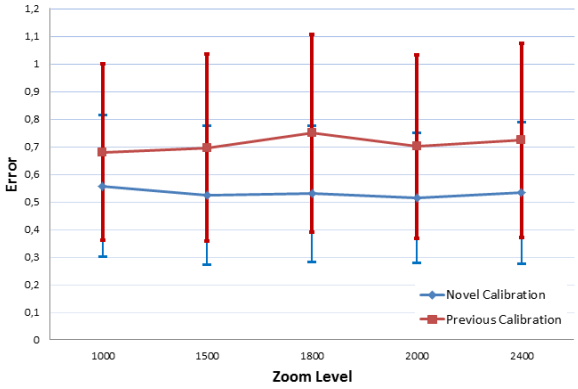

Figure 15: The data plotted represent the re-projection error made by using a traditional pattern (red line) and by adopting the proposed novel calibration pattern (blue line). The standard deviation is reported as error bars.

We evaluated also the time required for both the novel and the previous (used by the company) calibration procedures (see Fig. 1). The results described in Table 2 report the time difference of the calibration procedures in the soccer environment shown in the figures of this paper. Calibration time has been measured considering a within-subject design factor (the same human operator performed both the calibration approaches in the same operative scenario). The test has been carried out by an expert employee of the company, familiar with the previous method and properly instructed to use the novel one.

Multiple calibration runs have not been performed, because they would have required a different scenario, since making calibration with the same data multiple times would not provide relevant statistical evidence. Nonetheless, the difference in time is significant and the expert employee confirmed that other previous calibrations were in the same order of magnitude of the one reported in this table.

6 CONCLUSIONS

In this paper a novel approach for calibrating zooming cameras is presented. The method uses two novel calibration patterns, the former composed by squares and circles and designed for recovering the intrinsic parameters of the camera even for high zoom levels, the latter conceived for calculating the extrinsic parameters of

Time	Previous Method	Novel Method
In-lab calib.	2880 min.	30 min.
On-field calib.	90 min.	10 min.
Billboard Identification	10 min./bill.	0.5 min./bill.

Table 2: Calibration time using the previous calibration procedure and the novel one. The test has been carried out by an expert user.

the camera and for making the calibration process easier and faster.

A quantitative evaluation demonstrates: 1) The increase of accuracy of the proposed method for the intrinsic parameter estimation with respect to a traditional calibration pattern; 2) The decrease of the calibration time for the on-field calculation of the extrinsic parameters of the camera.

A practical application of the method for creating a virtual advertisement system has been described. The system is currently used by a company providing real time virtual advertisement services for sport events. The chosen application scenario is a challenging one, since a lot of changes in the zoom levels are required for adequately capturing sport events.

As future work we intend to analyse the problem of automatically individuating possible occlusion in front of the billboards that will be replaced.

ACKNOWLEDGEMENTS

This work has been realized within the project FILAS-RS-2009-1286 "Advanced 3D virtual advertisement system (Pixidis)" supported by Duel TV S.p.A. (`www.dueltv.com`) and FILAS Lazio (`www.filas.it`). The authors thank in particular Riccardo Colasanti, Antonio Di Noto, and Edoardo Egidi from Duel TV for their continuous support and for providing means and technical expertise to carry out this research.

7 REFERENCES

[Aga01a] Agapito, L., Hayman, E., and Reid, I. Self-calibration of rotating and zooming cameras. Int. J. Comput. Vision, vol. 45, pp. 107-127, 2001.

[AlA06a] Al-Ajlouni, S., and Fraser, C. Zoom dependent camera calibration. American Society of Photogrammetry and Remote Sensing, 2006.

[Dav03a] Davis, J., and Chen, X. Calibrating Pan-Tilt Cameras in Wide-Area Surveillance Networks. IEEE Int. Conf. on Computer Vision, pp. 144-150, 2003.

[Fan13a] FANN - Fast Artificial Neural Networks. http://leenissen.dk/fann/wp/

[Har04a] Hartley, R. I., and Zisserman, A. Multiple View Geometry in Computer Vision. Cambridge University Press, second edition, 2004.

[Kim00a] Kim, H., and Hong K. S. A Practical Self-Calibration Method of Rotating and Zooming Cameras. Int. Conf. on Pattern Recognition, vol. 1, pp.1354, 2000.

[May97a] Maybank, S. J., and Faugeras, O. A theory of self calibration of a moving camera. Int. J. of Comp. Vis., vol. 8, pp. 123-151, 1992.

[Oh09a] Oh, J., Nam, S., and Sohn, K. Practical Pan-Tilt-Zoom-Focus Camera Calibration for Augmented Reality, Int. Conf. on Computer Vision Systems, pp. 225-234, 2009.

[Ope13a] OpenCV (Open Source Computer Vision), http://opencv.org

[Sal02a] Salvi, J., and Armangué, X., and Batlle, J. A comparative review of camera calibrating methods with accuracy evaluation. Pattern Recognition, vol. 35, no. 7, pp. 1617-1635, 2002.

[Sarkis07a] Sarkis, M., Senft, C. T., and Diepold, K. Modeling the Variation of the Intrinsic Parameters of an Automatic Zoom Camera System using Moving Least-Squares. Int. Conf. on Automation Science and Engineering, pp. 560-565, 2007.

[Sin06a] Sinha, S. N., and Pollefeys, M. Pan-tilt-zoom camera calibration and high-resolution mosaic generation. Comput. Vis. Image Underst., vol. 103, n. 3, pp. 170-183, 2006.

[Sir04a] Sirisantisamrid, K., Matsuura, T., and Tirasesth, K. A simple technique to determine calibration parameters for coplanar camera calibration. TENCON, vol. 1, pp. 677-680, 2004.

[Stu97a] Sturm, P. Self-calibration of a moving zoom-lens camera by pre-calibration. Image and Vision-Computing, vol. 15, pp. 583-589, 1997.

[Tsa86a] Tsai, R. Y. An Efficient and Accurate Camera Calibration Technique for 3D Machine Vision. IEEE Conf. on Computer Vision and Pattern Recognition, pp. 364-374, 1986.

[Zhang00a] Zhang Z. A Flexible New Technique for Camera Calibration. IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 22, pp. 1330-1334, 2000.