Plug & Play: Fast Automatic Geometry and Color Calibration for Cameras Tracking Robots

Anna Egorova, Mark Simon, Fabian Wiesel, Alexander Gloye, and Raúl Rojas

Freie Universität Berlin, Takustraße 9, 14195 Berlin, Germany
http://www.fu-fighters.de

Abstract. We have developed an automatic calibration method for a global camera system. Firstly, we show how to define automatically the color maps we use for tracking the robots' markers. The color maps store the parameters of each important color in a grid superimposed virtually on the field. Secondly, we show that the geometric distortion of the camera can be computed automatically by finding white lines on the field. The necessary geometric correction is adapted iteratively until the white lines in the image fit the white lines in the model. Our method simplifies and speeds up significantly the whole setup process at RoboCup competitions. We will use these techniques in RoboCup 2004.

1 Introduction

Tracking colored objects is an important industrial application and is used in the RoboCup small-size league for locating robots using a video camera which captures the field from above. The two most important problems which arise in this kind of object tracking are: a) elimination of the geometric distortion of the cameras, and b) calibration of the colors to be tracked. It is not possible to calibrate geometry and colors manually, once and for all, since lightning conditions change from one place to another, and even from one hour to the next.

In this paper we describe the techniques we have developed for fast calibration of the global camera(s) used in the RoboCup small-size league. The paper is organized as follows. First we comment on related work. Then we describe in detail our new semi-automatic color calibration method and compare its results to the hand-optimized ones. The next section deals with the calibration of the geometric transformation of the field and compare the automatic with the manual method. Finally, we describe our future plans.

2 Related Work and Motivation

Zrimec and Wyatt applied machine learning methods to the color calibration problem [10]. They recognize regions delimited by edges and classify them according to features such as average hue, saturation, intensity, and others. A computer vision module for Sony robots uses those features to locate field landmarks.

Another approach for automatic calibration is to compute global histograms of images under different lightning conditions. Lookup tables for color segmentation are initialized in such a way as to make the new histogram equal to that found under controlled conditions [6]. In our case this approach would not work, since we do not use a single lookup table for the whole field. We recognize colors locally, using a color map which can change from one part of the field to the other. Some authors have tried decision trees in order to segment colors independently from light. However, they focus on object localization robustness and do not deal with the automation of the calibration [1].

Regarding the correction of the geometric distortion introduced by a camera, the canonical approach relies on determining first intrinsic camera parameters and then extrinsic parameters [2]. The intrinsic parameters can be measured in the laboratory. The extrinsic can be fitted by least squares, identifying points on the image with points in a geometric non-distorted model. We do not want to identify points on the field by clicking on them; the software should be able to automatically recognize the orientation of the camera and to select relevant points for matching them with the model.

Whitehead and Roth described an evolutionary optimization approach to camera auto-calibration [9]. Their method does not apply to our case, because they optimize the fundamental calibration matrix directly, without considering the radial camera distortion. Projective geometry methods alone solve one part of our problem, but not the whole problem. Some authors have studied real-time distortion correction for digital cameras, but without handling the projective transformation correction [4].

3 Semi Automatic Color Calibration

In [7] we described our use of color maps for robust color recognition of the robot's markers. They consist of a virtual grid superimposed on the field, one grid for each important color. At each grid node we store the RGB parameters of the color and the size of visible color markers for that part of the field, as seen by the camera. The grid must be initialized before starting to track objects, that is, we need an initialization step for each individual color. The color and marker size maps are further adapted during play, so that the tracking system updates the grid information whenever a change of illumination occurs.

3.1 Initializing the Color Maps

The global camera captures images in RGB format. This color space is not very useful when trying to achieve color constancy. Therefore, we operate in the HSV(hue/saturation/intensity value) color space. Fig. 1 shows the variability of the HSV components over the field as captured in our lab.

Fig. 2 shows the functional relationship between the intensity of the background pixels (green) and the color blobs on the field. The relationship is an

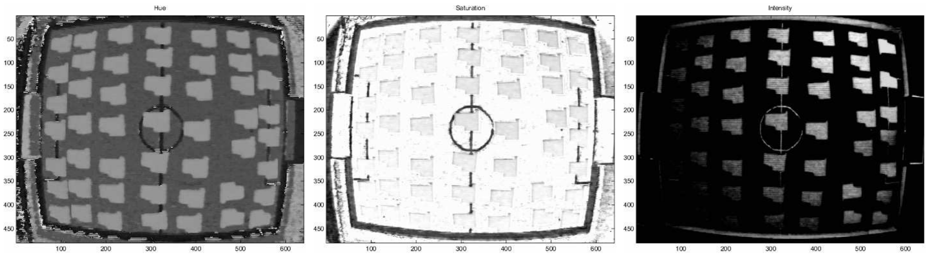

Fig. 1. The HSV components for the field of play with blue test blobs on it. The first image shows the hue component, the second the saturation, and the third the intensity. As can be seen, the intensity changes significantly throughout the field.

affine function, whose parameters can be computed from the background intensity and from two probes of the desired color positioned on the field. The function has two parameters: an additive constant and a scale factor. We compute both twice, from two probes, and average them for subsequent computations. More probes could also be taken, resulting in better averaging.

Given b_1 and b_2, the intensity of the background color at the point P_1 and P_2, and given c_1 and c_2, the intensities of the color C whose map we want to derive at the points P_1 and P_2, respectively, the translation factor t and the scale factor s are given by

$$t = \frac{(c_1 - b_1) + (c_2 - b_2)}{2} \qquad\qquad s = \frac{(\frac{c_1}{b_1}) + (\frac{c_2}{b_2})}{2} \tag{1}$$

When the background intensity b at a point P is known, the unknown intensity c of the color C at this point P is given by

$$c = bs + (b + t)$$

We also want to estimate the size of the robot markers (color blobs) around the field. The apparent size of the markers varies with lightning conditions and camera distortion. We assume that their size is proportional to the intensity of the blob. We estimate the marker size by interpolating the sizes of the samples, according to the intensity. One can also use the camera distortion to calculate the size of the robot markers according to it.

To reduce the influence of noise, a median color is computed for the marker and the background color at the picked point. The radius for the area around the blob is predefined with respect to the blob-object which is identified by this color (for example the orange ball is a little bit smaller than a team marker). The size of the median filter for the background color, for its color grid, is exactly the size of a grid tile [7].

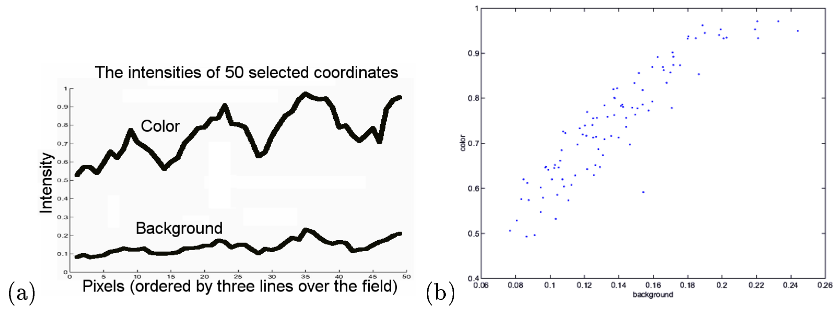

Fig. 2. The graph compares the intensity values of some random points on the green background and the color map. Illumination inhomogeneities produce different changes for different colors – scaling and translation factors are present: (a) shows the dependency of the intensities as a function of the field coordinates (three lines over the field); (b) shows the intensity of the color blobs as a function of the intensity of the background at the same points. The correlation is clearly visible.

3.2 Results and Comparison

In Table 1, the different forms of initializing the color map before starting a game are compared to the manual adjustment method, exemplarily for the ball. As shown, automatic initialization gives better results and smaller errors, compared to a uniform initialization of the color map. The most important improvement is the reduction of the maximum error, the relevant magnitude when trying not to lose track of an object in the field. The improvement in estimation of the ball size, as seen from the camera at different coordinates on the field, is also significant.

4 Automatic Geometric Calibration

The second camera setup problem we consider is the correction of the geometric distortion. Traditional methods require identifying pairs of points in the image and model space. One could artificially define those calibration points by using a carpet with marks, but that requires too much manual intervention. In this paper we show how to use the white lines on the field to determine the parameters of the transformation. The correlation between the extracted, and transformed lines, and the lines of the model, is a measure for the quality of the transformation. This measure enables us to find the best transformation with conventional optimization algorithms.

Our method is divided into the following steps: First, we extract the contours of field regions from the video image. Next, we find a simple initialization, which roughly matches the lines of the field to the model. Finally, we optimize the parameters of the transformation.

	Deviation from hand-optimized			
	Uniform		Automatic	
	maximum	mean	maximum	mean
hue	1.94%	0.85%	1.28%	0.61%
saturation	7%	1.4%	4.6%	1.7%
intensity	48%	9%	25%	9.4%
size	31 pixel	13 pixel	18 pixel	9.15 pixel
RGB distance	27.9%	5.2%	14.5%	5.7%
HSV distance	48%	8.7%	25.4%	9.73%

Table 1. Statistical results comparing initializations of color maps. The table shows the performance (relative to the hand-optimized color map) of a uniformly initialized color map and two different automatically initialized maps. The performance of one specific map is measured with the maximum and mean percentage deviation of computed values from those in the hand-optimized map.

4.1 Extraction of Contours

The non-white regions on the field are found by applying our region growing algorithm as described in [3]. Discrimination between white and non-white pixels is based on their relative intensity with respect to the local background. First, we average the intensity of a larger region around a given pixel, which is assured to contain a high percentage of background pixels. Relative to that rough approximation, we reject foreground pixels in a smaller region and locally determine the intensity from the remaining pixels. The contours of the regions found are the borders of the field lines (see Fig. 3).

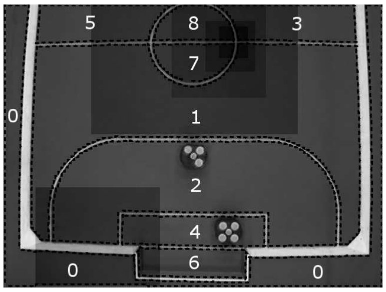

Fig. 3. F-180 Field Padova 2003, with artificial shadows with strong edges. Dashed lines show the field and out-of-field contours.

4.2 Quality Measure

Assume that we have a hypothetical geometric transformation T and that we want to measure its error. Since we cannot determine the exact location of a contour point p in the model, we approximate its error $E(T, p)$ with the distance between the transformed point $T(p)$ and the nearest white line in the model. For a set of points P, we compute the quality error $\hat{E}$ as

$$\hat{E}(T, P) := \bar{E}(T, P) + \sigma\left(E(T, P)\right)$$

where $\bar{E}$ is the mean error, and σ its standard deviation.

4.3 Initialization of the Transformation

To minimize the complexity of the optimization step, a smart initialization of the transformation T is useful. We assume that the field has only limited distortion, radially and in the perspective. A linear transformation is a good initial approximation, except for the alignment of the field. Finding a matching pair of rectangles in model and field let us compute such a transformation. In our case a bounding box around region 1 (see Fig. 3) results in a good approximation to a rectangle corresponding to one half of the field in the model. The alignment can easily be determined by testing all four orientations of the bounding box as initialization for the linear transformation. Using the quality measure described above for the contour of region 1, the best match identifies the orientation of the field, since the region is not symmetric with respect to rotation. The association between the field sides in image and model is arbitrary. The transformation T can be initialized accordingly to the linear transformation T_{init}. We use a biquadratic transformation as our standard transformation (see Section 4.4), but other transformation models are possible.

4.4 Optimization

We use for both coordinates in model space a biquadratic interpolation in a single 3×3 grid to determine the coordinates in model-space, giving us 18 parameters to be optimized. The initialization can be analytically derived from T_{init}. However, our approach relies only on the convergence properties of the transformation function and not on the function itself.

Given the transformation T, gradient descent with linearly falling step-length suffices to optimize the placement of the vertices with sub-pixel accuracy. To achieve faster convergence adaptive selection of step-length and/or conjugate-gradient methods can be used. Gradient descent only requires a non-degenerate initialization with the approximate orientation. In our experiments, even the whole image as rectangle for the initialization converged against the global minimum.

Other transformation-functions can be optimized similarly. Depending on its convergence properties, a global optimization method may be necessary, for example simulated annealing.

4.5 Calibration Results

We applied our algorithm to a set of real video images, captured two different camera-systems (Basler 301fc, Sony XC-555). We presented the algorithm with two different fields, the field of the Padova World Cup 2003, built according to the F-180 2003 rules, and our lab-field, which has slightly different dimensions and several rough inaccuracies. The images had different brightness, and some were artificially modified to simulate shadows with strong edges (as can be seen in Fig. 3), which normally are not present on the field and strain region-extraction

(a) (b)

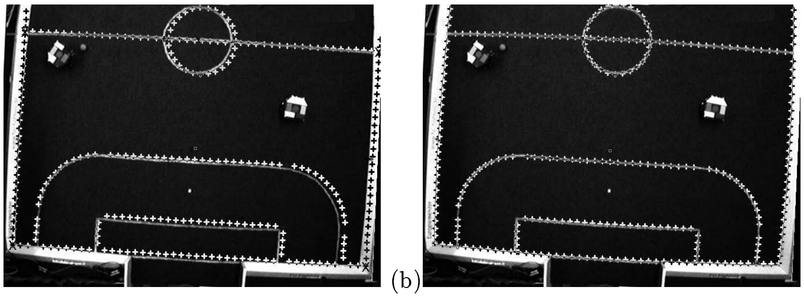

Fig. 4. The model matched on the rotated field (a) after the initialization and (b) after the optimization step.

more than the soft shadows usually observed. Furthermore, some images were rotated up to 20^o.

For a correct field model, our algorithm could adjust the geometric biquadratic transformation for all images presented, given the constraints on the distortion, without further adjustments to the algorithm after the initial selection of the Padova images. Subsequent attempts to improve manually the parameters of the transformation function resulted only in worse results, both with respect to the average and standard-deviation of the error.

In order to speed up the algorithm, initially only a subset of the contour points is taken into account for the quality measure. The amount is gradually increased as the error decreases. We use an adaptive step-width, which slowly increases, unless the gradient step increases the error. Then the step-width is reduced. The step-width is independent of the norm of the gradient.

We started the optimization with a step-width of 1 pixel and only using every 20th contour point in the measure. The optimization step required on our set of images at most 6 seconds (5 seconds with conjugated gradient) to adopt the parameters in 1/100 pixel accuracy and in average 4 seconds (4 seconds) on an Athlon XP -2400+ (2GHz).

5 Future Work

The color map for a specific color is initialized by clicking on two markers during setup. This takes a few seconds. In the future, our vision system will automatically detect colors differing from the background, and will initialize a color map for them.

For the geometric calibration, we want to evaluate other transformation-functions, which are probably more accurate, due to their better modeling of the distortion.

The improvements reported in this paper have the objective of achieving true "plug & play" capability. In the future it should be possible just to place robots on the field and start playing immediately against another team, whose colors and marker orientation will be determined automatically. This would speed RoboCup competitions significantly.

References

1. Brusey, J., and Padgham, L., "Techniques for Obtaining Robust, Real-Time, Colour-Based Vision for Robotics", *Proceedings IJCAI'99 - International Joint Conference on Artificial Intelligence, The Third International Workshop on RoboCup – Stockholm*, 1999.
2. Forsyth, D. A., and Ponce, A., *Computer Vision: A Modern Approach*, Prentice Hall., 1st edition, 2002.
3. von Hundelshausen, F., and Rojas, R., "Tracking Regions", in D. Polani, B. Browning, A. Bonarini, K. Yoshida (Eds.): *RoboCup-2003 - Robot Soccer World Cup VII*, Springer-Verlag, 2004.
4. Gribbon, K. T., Johnston, C.T., Bailey, D.G., "A Real-time FPGA Implementation of a Barrel Distortion Correction Algorithm with Bilinear Interpolation", *Image and Vision Computing*, Palmerston North, New Zealand, pp. 408-413, November 26-28, 2003.
5. Jacobsen, "Geometric Calibration of Space Remote Sensing Cameras for Efficient Processing", *International Archives of Photogrammetry and Remote Sensing*, Vol.32, Part I, pp. 33-43.
6. Kulessa, T., and Hoch, M., "Efficient Color Segmentation under Varying Illumination Conditions", *Proceedings of the 10th IEEE Image and Multidimensional Digital Signal Processing Workshop*, July 12-16, 1998.
7. Simon, M., Behnke, S., Rojas, R.: "Robust Real Time Color Tracking" In: Stone, P., Balch, T., Kraetszchmar (eds): *RoboCup-2000: Robot Soccer World Cup IV*, pp. 239-248, Springer, 2001.
8. Rojas, R., Behnke, S., Liers, A., Knipping, L.: "FU-Fighters 2001 (Global Vision)", In: Birk, A., Coradeschi, S., Tadokoro, S. (eds): *RoboCup-01: Robot Soccer World Cup V*, Springer, 2001.
9. Whitehead, A., and Roth, G., "Evolutionary Based Autocalibration from the Fundamental Matrix", in S. Cagnoni, Stefano Cagnoni, Jens Gottlieb, Emma Hart, Martin Middendorf, Gnther R. Raidl (Eds.), *Applications of Evolutionary Computing – EvoWorkshops 2002, EvoCOP, EvoIASP, EvoSTIM/EvoPLAN*, Kinsale, Ireland, April 3-4, Springer-Verlag, 2002.
10. Zrimec, T., and Wyatt, A., "Learning to Recognize Objects - Toward Automatic Calibration of Color Vision for Sony Robots", *Workshop of the Nineteenth International Conference on Machine Learning (ICML-2002).*