REAL TIME SOCCER FIELD ANALYSIS FROM MONOCULAR TV VIDEO DATA

M. Hoernig[1, 2], M. Herrmann[1, 3], B. Radig[1, 4]

[1]Image Understanding and Knowledge-Based Systems,
Technische Universität München, Boltzmannstrasse 3, 85748 München, Germany
[2]hoernig@in.tum.de, [3]herrmmic@in.tum.de, [4]radig@in.tum.de

Soccer games, respectively their video records, are an issue of extensive challenging research in image sequence processing. Topics of particular interest are player and field tracking. In this paper we will do an important step in this direction with a new approach to detect players and field lines in monocular TV video data. To achieve this goal, the field area and the grass colors are to be determined. This is done by involving contextual knowledge as well as a new method for color segmentation, which frame wise selects polyhedrons within the RGB cube. As an addition, the resulting algorithms are able to detect non field view images and work in real time.

Introduction

In this paper we show how robust field area and color detection can assist in the automatic real time analysis of soccer game videos. Our approach overcomes special challenges like multiple colored areas (e.g. due to bad weather conditions) or field color regions beyond the field. Furthermore the presented concepts are adaptable to other field based sport games.
The remainder of the paper is structured as follows; after discussing related work in section two, the novel algorithms are presented in sections three to five. Afterwards the results are shown in comparison with ground truth data in section six. The last section will give a conclusion and show possible applications.

Related Work

While Lu et al. [1] are using a computation time intensive edge matching approach for the field recognition, the usual approach for this fundamental step is color based. Often a simple pre-defined color with thresholds, e.g. Tong et al. [2] and Utsumi et al. [3], or peaks in a certain color space, e.g. Choi et al. [4], Ekin et al. [5] and Yu et al. [6], are used. The ASpoGAMo approach [7] is based on a Gaussian mixture model, but only able to detect single field color peaks automatically. Learned models, e.g. Huang et al. [8], can show weaknesses in bad image conditions. Mostly, the field is afterwards determined directly from points with the selected color key or using a region growing method as motivated by Yu et al. [6]. Beside the assumptions of field color hue and big field area, the usage of contextual knowledge is either low or not real time capable ([1]). In contrast, our system overcomes these weaknesses and is specified in the special context of soccer games, but yet adaptable under certain circumstances.

Field hull determination

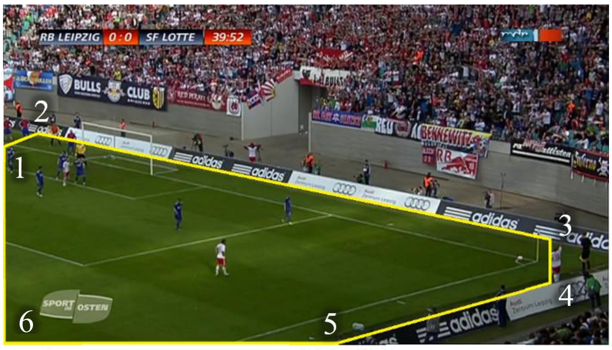

Fig. 1. The polyhedral field hull (yellow) with point naming. (Source: Mitteldeutscher Rundfunk)

To retrieve a polygonal hull (see fig. 1) we need two basic assumptions: the field is a rectangle, lying on the ground with the camera standing next to it, the lens directed on one of its inner points and a focal length resulting in an image that is mostly covered by the described field. The second assumption is that the field is green[2]. The field hull determination heuristic consists of the following steps:

1. Creation of binary hue mask H (HSV) through thresholding
2. Morphological selective opening of H:
    a. Erosion by a square (20px)
    b. Retrieve connected components
    c. Delete all components which are not the biggest one
    d. Dilate by a square (20px)
    e. Name the result as H'
3. Calculate contour C and smallest surrounding rectangle W parallel to the coordinate axes of H'
4. Six point polygon hull determination

Step 4 has to be discussed in more detail. The six points are arranged according to fig. 1. We start with point 1: This is, colloquially spoken, the topmost point where the field hull leaves the left line segment of W. The y-coordinate is determined by using the median height of five topmost vertical scan line intersections with C. The scan lines start from the left of W and are equidistant. The x-coordinate lies on the left of W. The same procedure is repeated mirrored for point 3. A temporary polygon T is now constructible, containing the points 1 and 3 and the bottom points of W. The next point in the processing order is the top point 2. Starting from the topmost point in C an optimization

$$\min_{p\in C}(\mathrm{Fill}(T \cup \{p\}) \vartriangle H') \qquad (1)$$

in a neighborhood of p is performed while $\mathrm{Fill}(T \cup \{p\})$ has to be convex!

The optimal p is added to T, which now contains three fixed points (1, 2 and 3). We propose to use this pentagon and cut everything beneath a line off so that the error region ($\mathrm{Fill}(T) \vartriangle H'$) is minimized. To reduce the number of line candidates, a small number of vertical scan lines can be chosen and intersected with the error region. All combination of two either top- or bottommost points within the connected components of the result can compose the line candidates.

Putting everything together the field hull is now complete. Results and quality evaluation are shown in section six.

Selecting field colors

The calculated hull allows an efficient and sophisticated grass color estimation by calculating an area within the RGB cube with the field describing colors. Because soccer games, like many other field games, are played on ground of uniform color, it is possible to speed up the estimation process drastically. As stated in [2], the intensity itself is unfeasible for the color detection. However, due to strong dependence of the color channels at field points (correlation in homogenous colored data), working on intensity images within the field regions offers a big complexity reduction combined with small and measurable data loss. This statement applies also in challenging conditions like shady or sodden environments. For simplicity, we use $V = \frac{1}{3}(R + G + B)$. The key task is to find the position of the color describing line in the RGB cube, select the appropriate segments and inflate the line in each axis by an automatically determined amount (see fig. 2). Nevertheless, the presented algorithm is also able to find distinct color areas (Note: $\mathbf{1} := (1, 1, 1)^{\boldsymbol{T}}$):

1. Smooth the input RGB image A with a small square mask (7x7 pixels)
2. Calculate the standard deviation image D of the result with same mask size
3. Store for each pixel in D the maximum value of all channels in D'
4. A pixel in D' is considered as homogenous if its value is below the sample mean of D'
5. A pixel in D' is considered as a field pixel if it is homogenous, within the field area and has an appropriate hue (for both see section 3)
6. All field pixels form region F
7. Use the relative histogram of $V(F)$ and select its biggest function values until their sum exceeds a predefined level (e.g. 98%) and merge the found entries into intervals, stored in interval set **I**
8. Calculate μ, the vector of the sample mean for the color vectors of A in F and μ_V the sample mean of the values of V in F; then the line shift is $\tau = \mu - \mu_V \mathbf{1}$
9. Get the sample variance matrix Σ in F of the vectors within the three channel

---

[2] [1] We used a hue value between 63° and 113° (HSV) within our test cases.

difference image $(A - \mu) - (V - \mu_V)\mathbf{1} = A - V\mathbf{1} - \tau$. The outer diagonal elements are negligible and set to 0

10. Iterate through A:
    a. Store the color components of the current pixel in a vector c
    b. $\lambda_\tau(c) := \frac{1}{3}(\mathbf{1}^T c - \mathbf{1}^T \tau)$
    c. Get $y = \underset{x \in \bigcup \boldsymbol{I}}{\operatorname{argmin}} |x - \lambda_\tau(c)|$
    d. The shortest Euclidean distance is: $d = c - \tau - y\mathbf{1}$
    e. Weighted: $d' = \sqrt{d^T \Sigma^{-1} d}$
    f. Threshold depending on d'

As depicted earlier, the basic idea is to work on a single shifted intensity line in the RGB cube. That line is described as

$$l: x = \tau + \lambda\mathbf{1}, \quad 0 \le x_1, x_2, x_3 \le 1. \qquad (2)$$

Now for every color c, a shortest distance from l to c (on a line orthogonal to l) can be constructed. The intersection is calculated in algorithm point 10b.
Interpreted geometrically the selected area in the cube is equal to a set of extruded ellipsoids (extruded polyhedrons in the discrete case). The extrusion axis is the shifted intensity domain l, the three ellipsoid half-axes are the standard deviations of the difference image (step 9). The number of connected components is given by the number of intervals in **I.** Usually the differences between the original and an one channel image with τ-shift are, within field points, below the threshold of perception. This, of cause, leads to small color regions in the cube. An example for the region selection is shown in figure 2.

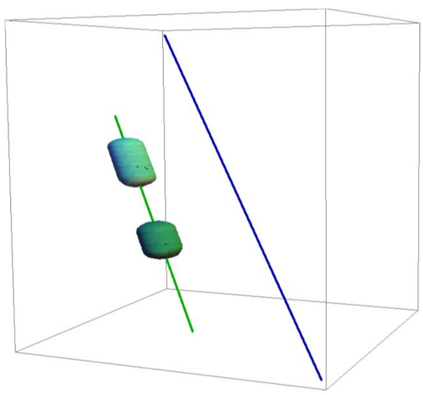

Fig. 2. The blue diagonal line is τ shifted and marked as a green line within the RGB cube. Afterwards the interval areas are inflated according the measured deviation values.

The threshold value in step 10f depends on the normalized Euclidean distance and is therefore weighted with the variances calculated in step 9. Hence, the limits can be expressed in terms of the standard deviation. An example for a mask created with the presented algorithm is shown in fig. 3.

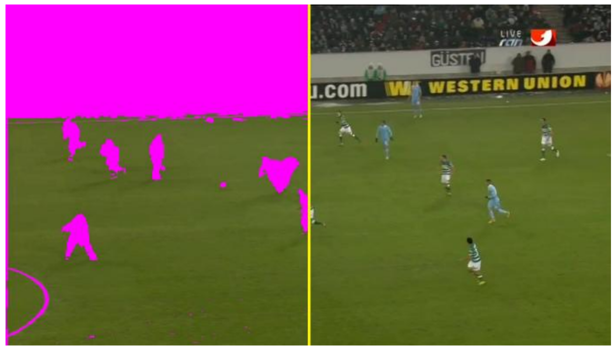

Fig. 3. The result of a binary field color calculation, non-field points are magenta shown at the left side (Source: ProSiebenSat.1 Media AG)

Refinement to continuous codomain

The approach discussed in the previous section allows to find the field colors and calculate binary masks. While this might be a sufficient result for a variety of scenarios, it leads to hard edges around inhomogeneous regions within the field (see fig. 4). This can be managed by relaxing the output to a mask with a continuous codomain.

Fig. 4. The grass mask (magenta) once created using a binary threshold (left), once refined to continuous domain (right).

The algorithm from section four changes only in step 10e where the error function

$$\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} \mathrm{d}t \qquad (3)$$

is implemented in a shifted form as

$$\operatorname{erfs}_{a,b}(x) := \frac{1}{2} - \frac{1}{2}\operatorname{erf}\left(\frac{\sqrt{\pi}(a+b-2x)}{a-b}\right). \qquad (4)$$

Using $0 < a < b$ as thresholds and thus as multiples of an “averaged standard deviation”, we are able to obtain masks with continuous codomain from $\mathrm{erfs}_{a,b}(d')$ like in fig. 4.: The first value, a, is the certainty value where every variance below will be accepted while the second, b, is the opposite, every color with higher distances will be declined (with a slight error). A distance between a and b is weighted with a swing. This results in better edge accuracy at inhomogeneous field regions than a linear function could offer.

Results and Evaluation

The proposed algorithms have been tested intensively on five different soccer games from diverse leagues and competitions and from various sources (different broadcasters, different resolutions). We created a ground truth database containing 54 images with annotated field and grass regions to evaluate the algorithms.

Table. Results for field and binary color detection (using a threshold of 6)

(Pixel wise error)	Precision	Recall
Field hull	99,31%	99,53%
Color area	99,11%	97,30%

We are also able to detect non field view images if a hull cannot be constructed or if the difference between hull and grass mask is too big. Both algorithms chained work in real time on a today’s standard PC.

Conclusion and Outlook

We present a novel approach for soccer field analysis. The field points and the hull can be estimated robustly with remarkable precision and recall rates in real time. In our current work, we combine both with a new line and person detection system to segment the image in its beneficial parts. Figure 5 and 6 show two examples. Future work will make use of this abilities, e.g. within a shot detection system with automatic field view identification for soccer games.

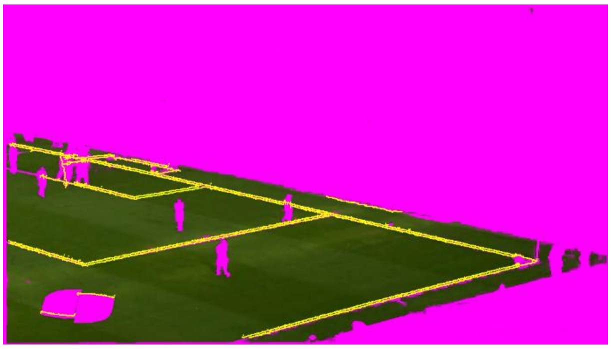

Fig. 5. The image from fig. 1 evaluated with the presented algorithms and an adapted line detector enables field position determination.

Fig. 6. Human and field color detection combined.(Source: ProSiebenSat.1 Media AG)

References

1. W-L. Lu, J.-A. Ting, J. J. Little, K. P. Murphy. Learning to Track and Identify Players from Broadcast Sports Videos // IEEE Transactions on Pattern Analysis and Machine Intelligence – 2013 – Vol. 35, No. 7 – P. 1704-1716
2. X.-F. Tong, H.-Q. Lu, Q,-S. Liu. An Effective and Fast Soccer Ball Detection and Tracking Method // Proceedings of the 17th International Conference on Pattern Recognition, 2004 – Vol. 4 – P. 795-798
3. O. Utsumi, K. Miura, I. Ide, S. Sakai, H. Tanaka. An object detection method for describing soccer games from video // IEEE International Conference on Multimedia and Expo, 2002 – Vol. 1. – P. 45-48
4. S. Choi, Y. Seo, H. Kim, K.-S. Hong. Where are the ball and players? Soccer Game Analysis with Color-based Tracking and Image Mosaick // International Conference on Image Analysis and Processing – 1997 – P.196-203
5. A. Ekin, A. M. Tekalp, R. Mehrotra. Automatic Soccer Video Analysis and Summarization // IEEE Transactions on Image Processing – 2003 – Vol. 12, Issue 7 – P.796-807
6. X. Yu, Q. Tian, K. W. Wan. A novel ball detection framework for real soccer video // International Conference on Multimedia and Expo, 2003 – Vol. 2 – P.II 265-268
7. S. Gedikli. Continual and Robust Estimation of Camera Parameters in Broadcasted Sports Games // TUM Dissertation – 2009 – P.86-91 - http://mediatum2.ub.tum.de/doc/649729/document.pdf