Player Number Recognition in Soccer Video using Internal Contours and Temporal Redundancy

Matko Šarić, Hrvoje Dujmić, Vladan Papić, Nikola Rožić and Joško Radić
Faculty of electrical engineering,
University of Split
R. Boskovica bb, Split, Croatia
msaric@fesb.hr, hdujmic@fesb.hr, vpapic@pmfst.hr, rozic@fesb.hr, radic@fesb.hr

Abstract: - Player identification is important task in sport video content analysis. In case of soccer video player numbers can be exploited to perform recognition. Jersey numbers can be considered as scene text and main problems in localization and recognition are low resolution, low contrast, color aliasing, unconstrained background etc. This paper proposed new method for player number localization and recognition. Firstly jersey regions are extracted using region growing algorithm. These regions are segmented using the component of HSV color space that best separates jersey and number area. Then, novel method for player number localization based on internal contours is proposed. False number candidates are discarded based on area and aspect ratio. Extracted numbers are enhanced using image smoothing and rotation normalization. We also propose algorithm that improves OCR recognition performance using temporal redundancy.

Key-Words: - player number recognition, soccer video, HSV color space

1 Introduction
Automated content-based video analysis of sports events is research area with high commercial potential. Annotation of video stream with respect to their semantic content allows efficient searching and retrieval of video material. Many efforts in sports video annotation have been made on important events (highlights) detection like goals or shots on goal for soccer video. The comprehensive survey of sports-related indexing and retrieval can be found in [1].

Text in video images is classified into caption text, which is artificially overlaid on the image, or scene text, which exists naturally in the image. Player numbers can be classified as scene text and recognition of this kind of text is generally more difficult compared with the general text detection. Problems are caused by variation of orientation, size, position, illumination, focus etc. A survey of image and video text information extraction can be found in [2].

Regarding player number recognition we can cite Ye et al. [3]. They proposed jersey number detection method where image is segmented using generalized learning vector quantization algorithm. Pixels are clustered into limited color-homogeneous regions in order to separate the jersey number from its background. Size and pipe-like attributes of digital characters are used to filter the candidates. Number recognition is preformed using K-NN classifier and Zernike moment features. In [4] algorithm proposed by Viola and Jones ([5]) is used for player numbers recognition. This algorithm is originally intended for face detection and it is based on cascade of classifiers. In [6] two-stage approach is proposed for player number detection: first step is extraction and selection of image corners using the Harris detector and second step is extraction of Maximally Stable Extremal Regions. These steps segment image in order to process it with OCR software. Region adjacency graph and picture trees are exploited by Andrade et al. ([7]) to perform object search using prior knowledge. Region analysis is further applied to the candidate regions to isolate player number which are then processed with OCR. In [11] same authors increase recognition count using the fact that multiple instances of the same player number exist across a number of frames.

We propose new method for player number recognition. First step is searching for regions of interest, i.e. jersey regions using region growing algorithm. These areas are segmented using component of HSV color space that best separates number and jersey. Several steps are done sequentially: players are extracted from segmented jerseys as internal contours. False candidates are then filtered using area and aspect ratio. Image smoothing and rotation normalization are employed before OCR processing. In order to enhance recognition performance we propose algorithm that uses temporal redundancy, i.e. the fact that same number must exist across a certain number of frames.

Rest of the paper is organized as follows. Section 2 discuses jersey region extraction and segmentation. Section 3. describes number candidates extraction using internal contours. Algorithm that improves recognition performance based on temporal redundancy is described.
in section 4. Results are presented in section 5. Conclusions are made in section 6.

2 Jersey Region Extraction And Segmentation

Segmentation of the whole video frame in order to find player number often results with very complex outcome. Prior knowledge about object (jersey and number color) can be used to reduce problem complexity. First step in our approach is jersey region extraction using region growing algorithm [12]. Criterion for seed point selection is Euclidean distance between frame pixel and jersey color in RGB color space. To be added to the jersey region, a pixel’s color should be close enough to the color of the seed point:

All extracted regions are potential player jerseys (figure 1b) and they have to be segmented in order to extract numbers. Rest of video frame is discarded from further processing. We propose segmentation method that exploits prior knowledge about team colors. Number and jersey usually significantly differ in one component of HSV color space to make number clearly visible in different conditions. Because of that segmentation is performed by thresholding that component. This procedure clearly separates jersey and number (figure 1c) as white object (jersey) with “hole” (number). Spatial relation between these two regions allows number extraction with internal contours detection. Segmentation is described as follows:

1. Using sample image define mean colors for jersey and number (denoted by the HSV vectors \( \mathbf{j} = [j_h, j_s, j_v] \) and \( \mathbf{n} = [n_h, n_s, n_v] \)). Pixel color is denoted by \( \mathbf{pixel} = [pixel_h, pixel_s, pixel_v] \)

2. Create binary image:

a) Number has achromatic color and jersey has chromatic color-segmentation by saturation thresholding

\[ h_{\text{diff}}(j_h, pixel_h) = \begin{cases} \Delta(j_h, pixel_h) & \text{if } \Delta(j_h, pixel_h) < 180^\circ \\ 360^\circ - |j_h - pixel_h| & \text{otherwise} \end{cases} \]

b) Jersey has achromatic color and number has chromatic color-segmentation by saturation thresholding

\[ \text{if } (\{j_h, < \lambda_j\}) \text{and } (\{n_h, > \lambda_n\}) \text{for each pixel} \]

\[ \text{if } (\{pixel_h, < \lambda_n\}) \text{and } (\{pixel_h, < \lambda_j\}) \]

\[ \text{if } (\{pixel_h, > V_{\text{thresh}}\}) \]

\[ \text{if } (\{pixel_h, < \lambda_n\}) \text{and } (\{pixel_h, < \lambda_j\}) \]

\[ \text{set pixel as black} \]

\[ \text{set pixel as white} \]

\[ \text{else} \]

\[ \text{set pixel as black} \]

\[ \text{set pixel as white} \]

c) Number and jersey has chromatic color-segmentation by hue thresholding

\[ \text{if } (\{(n_h, > \lambda_n\}) \text{and } (\{n_h, > \lambda_n\}) \text{and } (\{j_h, > \lambda_j\}) \text{and } (\{j_h, > \lambda_j\}) \text{for each pixel} \]

\[ \text{if } (h_{\text{diff}}(n_h, pixel_h) < \text{hue}_\text{thresh}) \]

\[ \text{set pixel as black} \]

\[ \text{set pixel as white} \]

\[ \text{else} \]

\[ \text{set pixel as white} \]

d) Number and jersey has achromatic color-segmentation by value thresholding

\[ \text{if } (\{n_h, < \lambda_n\}) \text{and } (\{n_h, < \lambda_n\}) \text{and } (\{j_h, < \lambda_j\}) \text{and } (\{j_h, < \lambda_j\}) \text{for each pixel} \]

\[ \text{if } (j_h, > V_{\text{thresh}}) \]

\[ \text{if } (\{pixel_h, < V_{\text{thresh}}\}) \]

\[ \text{set pixel as black} \]

\[ \text{set pixel as white} \]

\[ \text{else} \]

\[ \text{set pixel as black} \]

\[ \text{set pixel as white} \]
else
  if (pixel < V thresh)
    set pixel as white
  else
    set pixel as black

First step is system initialization that can be done after observing few seconds of video. HSV color space is applied to a wide range of applications since it clearly separates chromatic and intensity information, but use of this color space has three drawbacks: 1) hue is meaningless when the intensity is very low 2) hue is unstable when the saturation is very low 3) saturation is meaningless when the intensity is very low. Thus, our segmentation algorithm depends on mean saturations \( s_j \) and intensities \( v_j \) of jersey and number region.

In cases a) and b) segmentation is performed by saturation thresholding. In case a) segmentation is additionally refined by separating jersey from background using hue difference. In case b) intensity difference is used for same purpose. According to [2] saturation threshold is set to 0.2 because this value proved to be good tradeoff between color loss and matching of similar achromatic colors. Difference threshold for hue and intensity are empirically set to 20\(^\circ\) and 0.5.

In case c) most important cue to separate jersey and number is hue and image is segmented using hue difference. Case d) covers situation where jersey and number are distinguished by intensity difference. Typical example is black number on white jersey.

3 Number Region Extraction

Image segmentation algorithm described in previous section produces few segmented frame regions that are candidates for player jerseys. Spatial relation between number and jersey, that is fact that number is region surrounded by jersey region is exploited for number extraction. Jersey is represented as white object with black "hole" (number) and because of that number candidates can be extracted by internal contours detection (figure 1d) in previously segmented regions. Internal contours are found using algorithm implemented in OpenCV library [9]. In order to filter out non-number candidates area and aspect ratio of number bounding rectangle are used. Area is useful in eliminating small holes produced by segmentation process. Since it is assumed that player number often occurs in close-up shots we set area threshold to 0.06% of image area. Number bounding rectangle usually has height is larger than width. According to this fact height/width ratio threshold is set to 1.1. Finally remaining bounding

![Figure 1](image1.jpg)

\( a) \) Original image \( b) \) Jersey region candidates
\( c) \) Segmented jersey \( d) \) Internal contours
\( e) \) Number candidates after area and aspect ratio filtering
\( f) \) Number region cut from segmented image
rectangles (figure 1d) are used to extract number regions from segmented image (figure 1e).

To increase recognition performance enhancement of extracted number regions is required before OCR processing. OCR software is sensitive to character rotation. To solve this problem we used direction property of number region. This kind of region
descriptor makes sense in elongated regions only. Elongation is defined as ratio between the length and width of the region minimum bounding rectangle and most often number region can be considered as elongated region. If the shape moments are known, direction (theta) can be computed as defined in [10]:
\[
\theta = \frac{1}{2} \tan^{-1} \left( \frac{2\mu_{11}}{\mu_{20} - \mu_{02}} \right)
\]
where $\mu_{11}$, $\mu_{20}$ and $\mu_{02}$ are central contour moments. Then number region is rotated $\theta$ angle. Except rotation normalization number regions are also enhanced using median filtering.

Two digit numbers are represented as two numbers with spatially adjacent bounding rectangles. In order to correctly extract number distance between rectangles is checked. If this distance is low enough, two rectangles are framed together as new rectangle representing two digit number.

4 Temporal Redundancy

In order to enhance OCR recognition performance we exploited temporal redundancy, that is fact that same number is present across the certain number of frames. We propose algorithm that process OCR results and detect certain player number only if it is repeated certain number of times. OCR results are saved in text file in which one line contains numbers found in one video frame. Basic procedure is described as follows:

```
count[1..100]-array containing number counts
last[1..100]-last frame in which number appears
th1-threshold for number accepting
th2-distance threshold
for each line
    number=find number in text line
    count[number]++
last[number] == frame_counter
if (count[number]>th1) && (frame_counter-last[number])<th2
    output number
    count[number] = 0
    last[number] = 0
for i=1 to 100
    if (frame_counter-last[i])>20
        last[i] = 0
        count[i] = 0
frame_counter++
```

Our algorithm confirms OCR recognition of certain number when it appears more than $th1$ times. Distance in frames between two consecutive detections must be lower than $th2$ otherwise number counter is set to 0. This approach reduces number of false positive recognition (number is recognized when there is none or number is recognized wrongly). Thresholds $th1$ and $th2$ are empirically set to 5 and 20.

5 Results

We have tested proposed method on dataset composed of 2 soccer video clips in 640x480 and 640x352 resolution at 25fps. First clips contains 28280 frames (18:51 minutes) and second one 30150 frames (20:06 minutes). In this dataset player numbers in close-up shots appear 31 times through 1116 video frames in which number was detected by human observer. Two measures are used for method evaluation: recall and precision. They are defined as

\[
\text{Recall} = \frac{\text{Number of correctly recognized numbers}}{\text{Number of player numbers}}
\]
\[
\text{Precision} = \frac{\text{Number of correctly recognized numbers}}{\text{Number of recognized player numbers}}
\]

Results in table 1 are referred to video frames containing player number without using temporal redundancy. Test set contains 1116 frames showing player numbers. Results in table 2 are referred to shots containing jersey numbers, i.e. here we exploited fact that same number appear across the certain number of frames. Test set contains 58430 frames with 31 shots showing player number.

<table>
<thead>
<tr>
<th>Team</th>
<th>Correct</th>
<th>False</th>
<th>Missed</th>
<th>Recall</th>
<th>Precision</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>108</td>
<td>3</td>
<td>90</td>
<td>55%</td>
<td>97%</td>
</tr>
<tr>
<td>B</td>
<td>86</td>
<td>4</td>
<td>181</td>
<td>32%</td>
<td>96%</td>
</tr>
<tr>
<td>C</td>
<td>105</td>
<td>30</td>
<td>336</td>
<td>24%</td>
<td>78%</td>
</tr>
<tr>
<td>D</td>
<td>29</td>
<td>14</td>
<td>181</td>
<td>14%</td>
<td>67%</td>
</tr>
<tr>
<td>All</td>
<td>328</td>
<td>51</td>
<td>788</td>
<td>30%</td>
<td>87%</td>
</tr>
</tbody>
</table>
Table 1. shows that there are lot of missed numbers. This type of error is caused by non-rigid deformation of characters, blurred characters (move of players or camera), low contrast, variation in illumination etc. All these circumstances are very often found in our frame set because we select all frames containing number regardless of image quality. Significantly better results are obtained for precision because of lower number of false positive detections.

Results in table 2. show that use of temporal redundancy significantly improves recall rate (from 30% to 65%). Precision decrease is caused by fact that test set for performance in shots contain not only frames with numbers but all frames from video clips.

We found 4 papers dealing with player number detection and recognition ([3],[4],[6],[7]). In [4] and [6] results referred to shots are presented. Our approach achieves better recall and precision than method described in [4] (Recall=55%, Precision=55%). In comparison with [6] (Recall=67%, Precision=84%), our method has similar recall and weaker precision, but it must be emphasized that in [4] and [6] number recognition is performed only when face is detected while our test set includes all frames. This means that our system is more rigorously tested on false positive detections where number is present when there is none

Tests in literature are usually performed on datasets made of selected frames. In order to get objective method evaluation we include all frames from shots with player number (recognition in frames), that is all frames from entire video material (recognition in shots).

Because of this difference in datasets it is hard to directly compare our results with results presented by other authors.

5 Conclusion
A method for recognition of player numbers in soccer video based on temporal redundancy is proposed in this paper. Number region localization is performed in novel way: jersey region are extracted using region growing algorithm and segmented by thresholding component of HSV color space that best separates number and jersey area. This allows us to extract number regions by internal contours detection. Non-number regions are filtered out using area and aspect ratio of their bounding rectangles. Before OCR processing numbers are enhanced using rotation normalization and median filtering.

We also propose algorithm that exploits temporal redundancy and significantly increases recognition recall. This approach reduces negative effects of non-rigid deformation, blur, low contrast etc. and

ACKNOWLEDGMENTS

This work was supported by the Ministry of Science and Technology of the Republic Croatia under projects: ICT systems and services based on integration of information (023-0231924-1661) and Computer Vision in Identification of Sport Activities Kinematics (177-0232006-1662)

REFERENCES


---

**TABLE II**

<table>
<thead>
<tr>
<th>Team</th>
<th>Correct</th>
<th>False</th>
<th>Missed</th>
<th>Recall</th>
<th>Precision</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>5</td>
<td>0</td>
<td>2</td>
<td>71%</td>
<td>100%</td>
</tr>
<tr>
<td>B</td>
<td>4</td>
<td>0</td>
<td>2</td>
<td>67%</td>
<td>100%</td>
</tr>
<tr>
<td>C</td>
<td>9</td>
<td>6</td>
<td>4</td>
<td>69%</td>
<td>60%</td>
</tr>
<tr>
<td>D</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>40%</td>
<td>50%</td>
</tr>
<tr>
<td>All</td>
<td>20</td>
<td>8</td>
<td>11</td>
<td>65%</td>
<td>71%</td>
</tr>
</tbody>
</table>


