

Advanced Color Differentiation Approach for Basketball Hoops Detection

1st Petar Ristić
University of Niš
Faculty of Electronic Engineering
Niš, Serbia
petar.ristic@elfak.rs

2nd Miljana Milić
University of Niš
Faculty of Electronic Engineering
Niš, Serbia
miljana.milic@elfak.ni.ac.rs
ORCID: 0000-0001-7037-7709

3rd Dejan Madić
University of Niš
Faculty of Sport and Physical
Education
Niš, Serbia
madicdejan@yahoo.com
ORCID: 0000-0002-2978-2638

Abstract— Accurate basketball hoop detection is essential for automated score detection systems. This paper presents an advanced color differentiation approach to enhance hoop visibility in challenging scenarios, such as when both basketball hoop and a court are painted with the same color. The proposed method applies additional mathematical transformations to the input image, and ensures precise differentiation of the hoop from the background. The approach uses image color segmentation and edge detection techniques in order to reliably determine the coordinates of the hoop. The entire score detection system is based on Raspberry Pi 5 platform. Experimental results demonstrate the effectiveness of this approach.

Keywords—digital image processing, computer vision, IoT, edge detection, Raspberry Pi, neural network

I. INTRODUCTION

Accurate score measurements have played an important part in the evolution of sports by assuring fairness, consistency, and competition integrity. In basketball, precise scorekeeping has been vital since the sport's invention in 1891, that enabled objective game outcomes and performance analytics [1]. Over time, advancements in technology, such as digital scoreboards, instant replay, and player tracking systems, have further improved the accuracy and efficiency of measuring instruments and standards, that further minimized human errors and enhanced the spectator experience [2]. Reliable score measurements also play an important role in player evaluations, strategic decision-making, and historical record-keeping, in both professional and amateur levels of the game.

Computer vision techniques have attracted significant attention in sports analytics. These techniques have been used in basketball games to recognize objects, classify events, plan visual search tactics, make decisions, and analyze shooting motion trajectory [3]. Advances in computer vision have led to a greater knowledge of the game and player performance, and enabled more effective training approaches and strategies for teams and coaches. It would be impossible to analyze basketball motions, team strategies, and individual performance without ball and player detection and tracking devices and procedures.

This study proposes an improved color-based differentiation technique to enhance the visibility of basketball hoops in visually complex environments, particularly when the hoop and court share similar colors. The method incorporates additional mathematical transformations on the input image to achieve accurate separation of the hoop from the background. It employs color segmentation and edge detection algorithms to reliably

identify the hoop's coordinates. The complete score detection system is implemented on different Raspberry Pi platforms.

In the following we will first, explain in short, the Automatic Basketball Score detection system architecture (Koshko) [4]. Within the system a problem of basketball hoop detection occurs when both the hoop and the basketball playing court are painted with the same color. This requires additional image processing techniques that will be explained afterword. Then, Python-based image manipulations used to overcome this problem, together with the result of image processing are explained. The paper ends with the concluding remarks.

II. SYSTEM SETUP AND PROBLEM FORMULATION

Automatic basketball score detection system is a part of a larger Intelligent computer vision system for Smart IoT

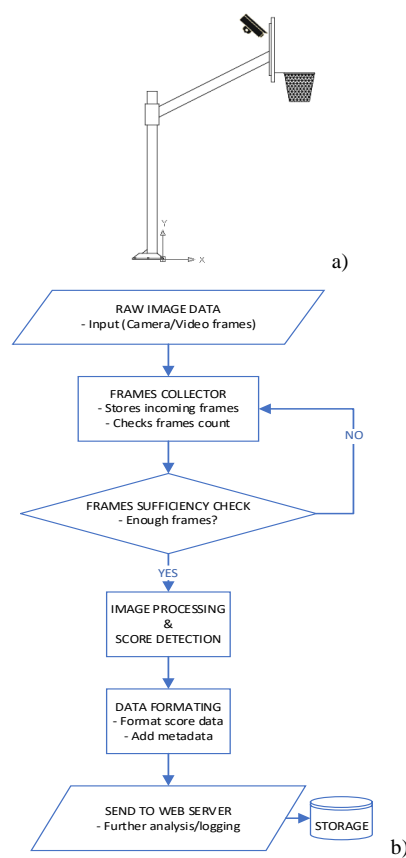


Fig. 1. Computer vision-based score detection system for Smart IoT kernebasketball

basketball. Score detection subsystem consists of a single camera placed behind the backboard, slightly above the basketball hoop. The camera protective case is large enough to put a Raspberry Pi based score detection unit. This ensures low possibility of damage and low cost, and is shown in Fig. 1a). With this configuration, additional court infrastructure is not required.

The architecture of an intelligent computer vision-based score detection system for Smart IoT basketball is shown in Fig. 1b). This block diagram illustrates paths of data within the system. First, we need to obtain raw image data as an input for the data processing. The image must be of a correct format for the system to properly and efficiently process it. Obtained input data could be processed and sent to the Score detection block, from which final data can be formatted for uploading to a web server (using JavaScript Object Notation – JSON format).

A. Problem Formulation

For score detection we use an algorithm based on raw image data processing. First, we identify the hoop's coordinates. Then we determine if the center of the ball has passed through the hoop. A trained neural network detects the ball. A score point is detected and incremented when the size of the ball reduces while its center falls within the hoop. These are two indicators that the ball has passed through the hoop. This is illustrated in Fig.2.

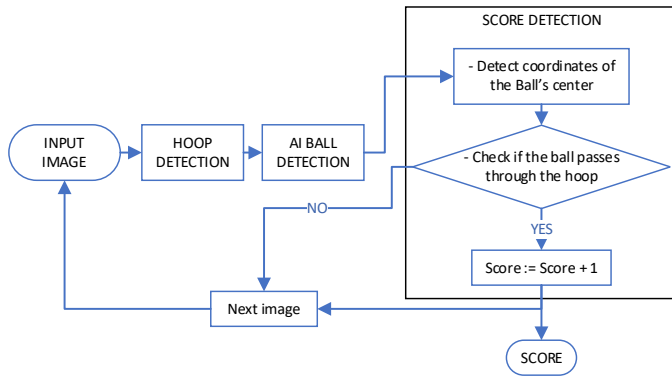


Fig. 2. Score detection phases

In order to indicate a score point, it is essential to correctly identify the coordinates of the hoop. The issue arises when an overlap of two objects (i.e. the hoop and the basketball court floor) that have similar color occurs. It might be difficult for a low-quality camera to distinguish between them and they can often be treated as a single object. In order to solve this problem additional data about the color nuances of objects must be extracted to distinguish between the hoop and the court. Auxiliary noise that resembles a basketball hoop, such as a drawing of a hoop on the court, could also exist. This is trivial, but it could cause problems and requires extra caution in processing.

III. ADVANCED IMAGE PROCESSING

To process the image, we employ a combination of filtering operations, morphological transformations, and masking techniques. The ensure system flexibility for different image sizes and qualities we use different kernels in operations and transformations of matrices. In terms of image processing algorithms, a kernel is a small matrix (typically 3×3 , 5×5 , etc.) used to modify an image through various operations such as filtering, edge detection, blurring, and morphological transformations [5]. A kernel is convolved over the image—

meaning it moves across it, element by element, and applies mathematical operations to the pixels within that small window. The results of these operations generate a new, filtered version of the image.

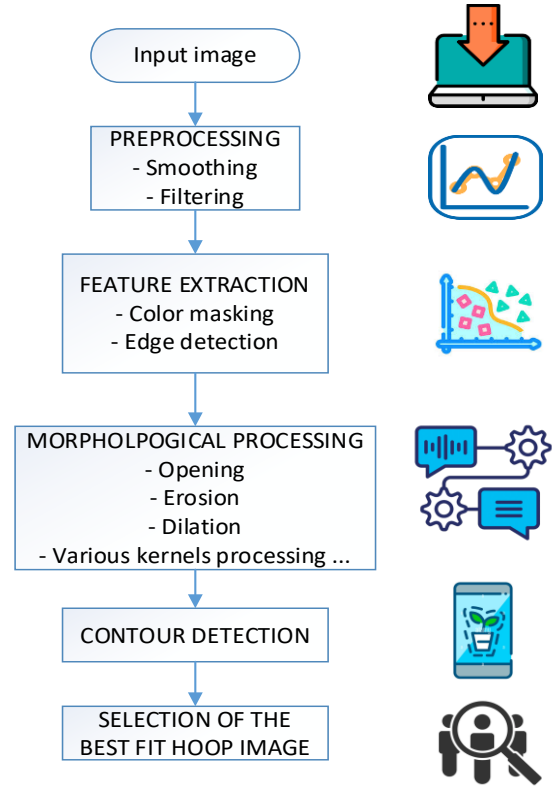


Fig. 3. Image processing procedure

The image processing flow for detecting a basketball hoop within a complex image consists of multiple stages, beginning with a preprocessing, i.e. a series of smoothing operations that reduce noise and preserve critical structural details [6]. Initially, bilateral filtering is applied to the input image. This technique smooths homogeneous regions and ensures that significant contours are not lost in the process. Gaussian blur with a small kernel is then added to further reduce local fluctuations without sacrificing the sharpness of those critical edges. This two-step phase prepares the image for further feature extraction.

Once the image has been suitably preprocessed, procedure continues with the feature extraction i.e. edge detection. The Canny edge detection technique is used because it is good at recognizing boundaries via gradient analysis [6]. Here, Sobel filters are applied to compute intensity gradients in both horizontal and vertical directions, and highlight the strong transitions in pixel values that often correspond to object outlines [6]. This phase is especially important for distinguishing the basketball hoop from similar-colored items in the environment, such as an orange court surface.

Using the extracted edge information, the color-based segmentation is performed. The image is converted to the HSV color space, which separates chromatic content (hue) from intensity (value). This conversion allows more precise color discrimination. Within this color space, two color-based masks are defined to capture the expected appearance of the hoop: M_{orange} - targets orange-yellow hues (main hoop color), and M_{red} - it captures red hues (edge transitions or faded red parts).

These masks address the variability in hoop coloration under different lighting conditions or from different viewing angles.

Two supplementary lighting and darkness masks address illumination extremes. M_{light} mask identifies brightly lit regions and reflections, which often introduce artifacts that must be accounted for, while M_{dark} mask targets dark regions that do not contribute useful information for hoop detection and are therefore excluded from consideration.

The resulting image, which integrates both color and edge information, represents a refined approximation of the basketball hoop's position. This final image can be expressed as:

$$I_{comb} = (M_{orange} + M_{red}) \oplus M_{light} \oplus (M_{dark} \cdot M_{orange}). \quad (1)$$

To enhance spatial coherence, the previously detected edge map is slightly dilated to increase the continuity of detected boundaries. This enhanced edge map is then joined with the combined color mask through a bitwise AND operation. As a result, a preliminary location of the basketball hoop is obtained as (2):

$$I_{final} = I_{comb} \cdot |\nabla I|. \quad (2)$$

Here, I_{comb} represents the combination of all constructed simple masks constructed with the basic color masking techniques, while, I_{final} is obtained by overlapping the combined color mask and the edge filter. ∇I is calculated as the edge gradient of the image using Sobel's kernels for the vertical and horizontal direction. Sobel's kernels are given as:

$$S_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, \quad (3)$$

$$S_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}. \quad (4)$$

Further improvement of the image is necessary to isolate the hoop from common visual disturbances such as the net or stray reflections. The following morphological filtering operations are applied [7]:

- Gabor-based opening in two orientations: $\frac{\pi}{4}$, and $-\frac{\pi}{4}$, to suppress net patterns (Fig. 6),
- Square kernel opening, to clean up isolated noise,
- Erosion with Gaussian kernel (2 iterations) to reduce minor artifacts (Fig. 7), and
- Dilation to reinforces edges and prepare for final contour extraction.

All these steps provide a robust foundation for detecting the basketball hoop in diverse visual conditions.

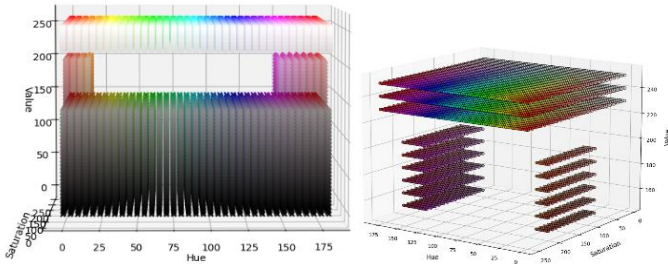


Fig. 4. Color spectre before and after masking logic operations

Visualization of all applied kernels are given in Figures 5 to 7.

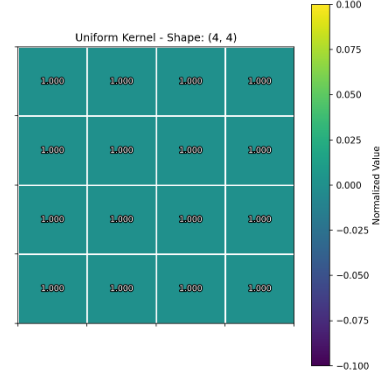


Fig. 5. Box filter used in this paper, 4x4 square kernel

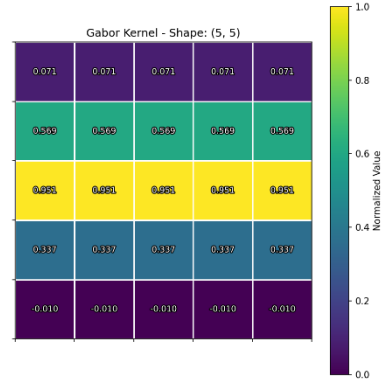


Fig. 6. Gabor kernel, 5x5 with varying degrees of rotation

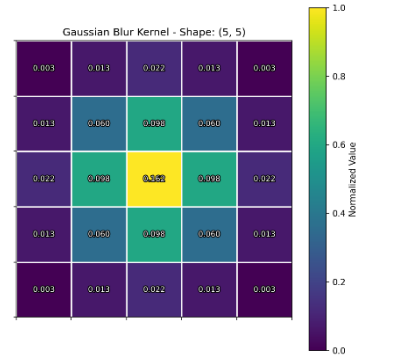


Fig. 7. Steep Gaussian kernel, 5x5

IV. RESULTS

After implementing the described calculations on a Raspberry Pi 5 platform, the system was tested for functionality. Intermediate results obtained after particular phases of image processing are shown in Fig. 8, and Fig. 9.

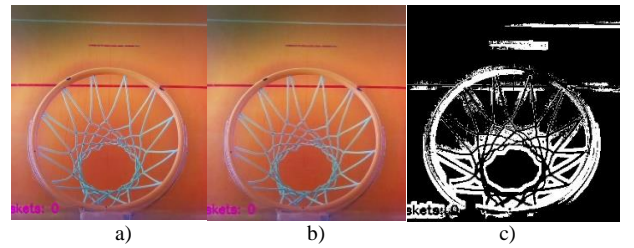


Fig. 8. a) Original image b) image after smoothing c) image after color masking

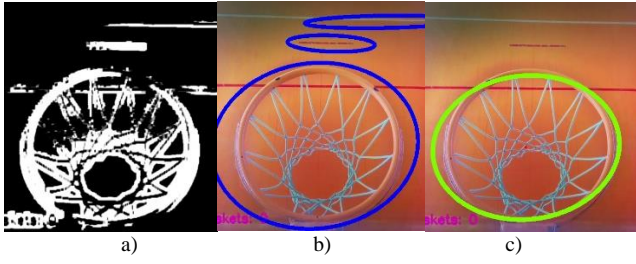


Fig. 9. a) Image after morphological operations b) finding all valid ellipses in the masked image c) selection of the correct hoop ellipse

We measure the accuracy of hoop calculation achieved by the algorithm in different scenarios (high luminescence on the rim, different camera angles and various background complexity), and the speed of processing for different image sizes and Raspberry Pi family devices.

The accuracy of calculations was assessed by evaluating the deviation from the center of the ellipse and determining the percentage of the hoop covered by our calculations. As for the calculation of the center of the ellipse for a set of 20 images we have obtained the MSE (Mean Squared Error) of 3.56%. Besides, the calculated hoop ellipse and the real image of the hoop achieves 94.67% overlap.

TABLE I. RASPBERRY PI PERFORMANCE

Processing time for varying resolutions on different edge devices [ms]			
Resolution	Raspberry Pi 5	Raspberry Pi 4	Raspberry Pi Zero 2
640x480	110.91	217.26	245.30
1920x1080	509.27	1124.36	11254.14
500x536	116.41	181.27	557.37
400x400	95.39	147.67	370.42
1920x1134	572.13	1230.07	17621.23

TABLE II. RASPBERRY PI COST EFFICIENCY

Cost efficiency for different resolutions and different edge [ms/\$]			
Resolution	Raspberry Pi 5 (\$50)	Raspberry Pi 4 (\$35)	Raspberry Pi Zero 2 (\$15)
640x480	2.22	6.21	16.35
1920x1080	10.19	32.12	750.28
500x536	2.33	5.18	37.16
400x400	1.91	4.22	24.69
1920x1134	11.44	35.14	1174.75

As shown in Table I we can conclude that the fastest images processing is obtained using the Raspberry Pi 5 platform, but if we consider the price of these micro-computers we should note

that Raspberry Pi 4 gives the best price-to-performance ratio due to lower price in compare to Raspberry Pi 5. It should be noted that one frame is enough to perform the entire hoop detection process, since the hoop itself is a static object on each frame. Including more frames into the computation would be unnecessary, and would require more power and time without the actual need.

From Table II we can deduce that the most advanced development board (the most expensive i.e. Raspberry Pi 5) offers highest computation efficiency.

V. CONCLUSIONS

The image processing approach described in this paper demonstrates a robust procedure for detecting a basketball hoop by combining edge information with targeted color segmentation in the HSV space. Through careful smoothing, edge detection, and morphological filtering, the method effectively isolates the hoop from visually similar backgrounds and common artifacts such as reflections and net structures. Overall, the approach provides a reliable foundation for real-time score detection basketball-related vision systems.

To enhance the accuracy of hoop location, techniques such as localization and image segmentation could be employed to mitigate issues like mask spillage and overshooting. Additionally, capturing video footage in color formats such as SRGB could be beneficial, as it broadens the range of colors available for processing.

ACKNOWLEDGMENT

This work was supported in part by the Ministry of Science, Technological Development and Innovation of the Republic of Serbia.

REFERENCES

- [1] M. Kocić, and D. Berić, Basketball (in Serbian), University of Niš, Faculty of sport and physical education, 2015.
- [2] D. Madić, M. Nikolić, and D. Stojiljković D. Madić, Measuring instruments in sports, physical education and recreation (in Serbian), University of Niš, Faculty of sport and physical education, 2015.
- [3] I. Ćirić, M. Milošević, N. Ivačko, A. Cvetković, M. Pavlović, and D. Krstić, "Intelligent Computer Vision System for Score Detecton in Basketball," Facta Universitatis, Series: Automatic Control and Robotics vol. 22, no. 2, pp. 75 – 85, 2023.
- [4] -, "Koshko", accessed on 9th of April 2025, at: <https://kosko.rs/kako-je-pocelo/>
- [5] R. Gonzalez, and R. Woods, Digital Image Processing (4th Edition), Pearson, 2018.
- [6] S. Sangwine, and R. Horne, The colour image processing handbook. Springer Science & Business Media, 1998.
- [7] E. Dougherty, and R. Lotufo, Hands-on morphological image processing, vol. 59. SPIE press, 2003.