8BallPool video analysis

Michele Sprocatti[1], Alberto Pasqualetto[2], Michela Schibuola[3]

Department of Information Engineering, University of Padua
Via Gradenigo 6, 35131 Padova, Italy
{michele.sprocatti[1], alberto.pasqualetto.2[2], michela.schibuola[3]}@studenti.unipd.it

Introduction

This is the report of the final project for the Computer Vision course, which goal is to develop a computer vision system for analyzing video footage of various "Eight Ball" billiard game events.

How the work has been split

The three group members collaborated by dividing the work. Each member was responsible for producing specific files, all of which were annotated with their respective author.

Split of the work

The source code reflects a division of labor across three key areas:

1. Detection and segmentation of balls and the playing table: this area, along with the main program functionality, was handled by Michele.
2. Metrics calculation and tracking: responsibility for this area fell to Alberto.
3. Transformation code and mini-map management: this area was overseen by Michela.

Working hour per member

The approximate number of working hours for each member of the group are these ones:

1. Michele: 40 hours.
2. Michela: 40 hours.
3. Alberto: 40 hours.

Elements of our project

Table detection

A mask-based approach was implemented for table detection, exploiting the consistent central positioning of the table within the dataset. The mask was generated by identifying the most common color in the image's central columns. Building upon this initial step, Michele exploits the Canny edge detector and OpenCV's HuoghLinesP function. Then the function analyzes intersections and merges nearby points, ensuring the consistent identification of four corner points corresponding to the table's corners in the processed dataset.

Table segmentation

In order to isolate the table with high precision, Michele employed a two-mask combined with k-means algorithm. The masks involved are:

1. Color-based masking: A mask was created that identified pixels corresponding to the table's color.

2. Corner detection masking: A separate mask was generated to capture the table's geometric features starting from the corners already detected.

By combining these two masks with the output of a k-means clustering algorithm with two clusters, the table was effectively isolated from the background.

Balls detection

To detect balls, Michele proposed a multi-step preprocessing approach. Initially, the table region was isolated by constructing a polygon using its corners and a color-based mask is generated. Subsequently, pixels outside the table were nullified, and k-means clustering was applied to the image. The resulting clusters were converted to gray-scale for Hough Circle Transform application. Circle parameters, such as radius and center color, were analyzed to identify potential ball regions. By calculating the mean radius of in-table circles with center not selected by the color mask, a radius range was established. Circles within this radius range were considered for further analysis. Ball classification involved creating a circular mask, computing the gray-scale histogram, and excluding background pixels from the values of the histogram. Peak values in the histogram were used to differentiate between striped and solid balls, while HSV color space analysis is used to distinguish white and black balls. After finding the balls, the team identified an optimization opportunity. Since there's only one white ball and one black ball, Michele implemented non-maxima suppression for white and black balls independently, in order to improve performance. The result of the detection process is then used to segment the balls.

Tracking

Mini-map creation

Video creation

Results

Table detection exhibits very high accuracy, in particular for each initial frame of each video four corner points are consistently identified across the dataset, the assumption that we made is that the camera does not move during a single clip so once the table is detected in the first frame we can use that information for all the frames of the same video. In contrast, ball detection is influenced by k-means clustering. To achieve consistent and satisfactory results, a fixed random seed is incorporated into the code. This method results in an average mAP of 0.72 for the dataset.

Qualitative results

Quantitative results

Conclusions