

Tracking player movements in a basketball game

The aim of this problem is to track basketball players in the given videos. Video is basically a recording of moving visual images. In order to detect objects that are moving in a video, we can compute the difference between two consecutive frames, for all frames. The objects that haven't moved at all will result into zero difference whereas the objects that have undergone some amount of displacement will produce a non-zero difference. This idea can be used to track players in the video at hand. The assumption being made for this idea to work is that the camera location must be constant throughout the recording and this is true in case of the given videos. Using this idea, I have implemented the following algorithm with the help of OpenCV in C++.

1. Play the video
2. Capture 2 current consecutive frames at a time (i.e. frame_1 and frame_2)
3. Convert them into gray scale images
4. Compute the absolute value of the difference between these 2 grayscale images
5. Perform smoothing and noise removal by applying Gaussian blurring operation to the difference image
6. Convert this difference image into a binary image by thresholding it at a specific level
7. This binary image contains information about the objects that have moved and the basketball is one of these objects
8. To retain only the players, perform morphological operations of dilation and erosion to get the refined thresholded binary image
9. Obtain contours of the objects in this binary image and determine objects having 4 largest contour sizes corresponding to the 4 players respectively. (Note: 2 largest contours for 2 players).
10. In order to avoid false positives, check if the area, width and height of the bounding rectangles around these contours is above the corresponding thresholds.
11. Draw bounding rectangular boxes around these objects at the same corresponding locations in current frame_1
12. Display this same frame_1 containing bounding rectangles around objects that have moved along with their locations
13. Move to the next two frames of the video and assign those frames to frame_1 and frame_2
14. Go to step 2 and carry the same process until the last frame of the video has been reached
15. By successively displaying the images assigned to frame_1 with their corresponding bounding boxes, a video gets created which basically shows how the player's movements are being tracked