Approach

In order to detect motion in the basket region, I extracted the first frame of the video and stored it as a reference. Every subsequent frame is compared with the first frame to detect motion which is essentially any change in the pixel values. This is achieved by taking the difference between the two frames and applying a binary threshold to the difference image. All frames are converted to grayscale and a Gaussian smoothening filter is applied to average pixels of given size before difference image is calculated. Morphological erosion and dilation are performed to get rid of unwanted jitters. In order to accentuate the difference in the region where the basketball is, I manually selected pixels corresponding to the basket and performed dilation. This was added to the rest of the image. A counter starts accumulating pixel values if the number of pixels with 255 value is greater than 500 (found by trial and error) and stops the accumulation if no 255 pixels are found. This accumulated value is compared for hit and miss activity and is found to be larger for hit. The range of values are found by trial and error. To avoid false hit or miss due to bouncing back of the ball from the basket rim, I have added a frame count condition before registering hit or miss.

If I had more time, I would collect more data to better optimize the algorithm. The given video misses some edge cases which can lead to misinterpreting hit or miss activity. Also, it is possible to detect the basket using an already trained model to label the pixels or we can train a model to do so. Also, if there is more data, a Recurrent Neural Network such as LSTM can be trained using video clips to activity recognition. This would make the model robust provided the model developed is good.