```
while (true)
    {
        Mat imgOriginal;

boolbSuccess = cap.read(imgOriginal); // read a new frame from video



if (!bSuccess) //if not success, break loop
        {
cout<< "Cannot read a frame from video stream" <<endl;
break;
        }

  Mat imgHSV;

cvtColor(imgOriginal, imgHSV, COLOR_BGR2HSV); //Convert the captured
frame from BGR to HSV

  Mat imgThresholded;

inRange(imgHSV, Scalar(iLowH, iLowS, iLowV), Scalar(iHighH, iHighS,
iHighV), imgThresholded); //Threshold the image

  //morphological opening (removes small objects from the foreground)
erode(imgThresholded, imgThresholded,
getStructuringElement(MORPH_ELLIPSE, Size(5, 5)) );
dilate(imgThresholded, imgThresholded,
getStructuringElement(MORPH_ELLIPSE, Size(5, 5)) );

  //morphological closing (removes small holes from the foreground)
dilate(imgThresholded, imgThresholded,
getStructuringElement(MORPH_ELLIPSE, Size(5, 5)) );
erode(imgThresholded, imgThresholded,
getStructuringElement(MORPH_ELLIPSE, Size(5, 5)) );

  //Calculate the moments of the thresholded image
  Moments oMoments = moments(imgThresholded);

double dM01 = oMoments.m01;
double dM10 = oMoments.m10;
doubledArea = oMoments.m00;

  // if the area <= 10000, I consider that the there are no object in
the image and it's because of the noise, the area is not zero
if (dArea> 10000)
  {
   //calculate the position of the ball
intposX = dM10 / dArea;
intposY = dM01 / dArea;
```