HW07A

SANDHYA MURALI
sm2290@g.rit.edu

ALGORITHM:

1.Read the image and convert it to double.
2. Access the red channel of the image
3.Apply median filtering for noise removal
4.Apply graythresh and select appropriate threshold
5.Convert the image to black and white based on the threshold
6.Apply erosion,closing and opening to separate the dice if touching each other
7.Use bwlabel to count the number of dice present
8. Iterate from 1: number of dice
    8.1. Detemine single region of dice
    8.2. Use regionprops to create a Convex Hull
    8.3 Determine the set of x coordinates and y coordinates from the ConvexHull
    8.4 Plot this Hull to check if the black dot comes inside the dice portion
    8.5 Take the inverse of the region
    8.6 Use bwlabel to compute the dots
    8.7 Print individual components

DISCUSSION:

Based on the algorithm coded above,dots on the individual dice is computed and displayed for most of the images.However if there is a motion blur,the dots on the image appears blured .Therefore this is almost equivalent to white and it considers as if no dots are present on the dice.Hence number of dice prints correctly.However it does not detect any dots.This can be improved if contrast and intensity is improved.Also some images of shadow changes,more light appears in the image.Hence more white dots are present and more number of unknown values get printed.Therefore maybe if we apply more efficient morphological operations and noise removal techniques.Also the number 6 on the dice,sometimes does not print correctly.This happens because either the dots merge to form single dot giving incorrect results.This can be corrected if we use better dilation techniques to separate the black dots.