Manaal Rehman
Student Number: 1602166

My approach to this task was in two steps. I applied thresholding to the image to segment out the blue and red blocks using their respective hue saturation values. I created a Gaussian filter to reduce any noise that was present in my image and then created an image matrix of hue saturation values with this filter, and converted it to a hsv format. I created binaries, which contained the result of thresholding either the blue or red pixels in the image (that were within the red or blue HS ranges). To clarify the image shapes I used 'bwmorph', this used Morphological structuring to tidy up the image. I used region properties to define the area in my binaries as regions. And by counting the regions I counted the number of block of each colour. I set these values as numA for blue and numB for red respectively. I used the following source as a guideline for colour segmentation: https://stackoverflow.com/questions/28004426/segment-pixels-in-an-image-based-on-colour-matlab The feature I used to obtain the correct size of the block was the geometric ratio between width and length, this would be independent of the scale of the image. For all of the regions I obtained both the Major and Minor Axis Length and calculated the ratio between them. If the ratios were not within a certain range, I reduced numA or numB by one. (Meaning that a region was not of the correct size to be of either a red 2 by 2 block or a blue 2 by 4 block). I worked out the ratio, by finding the minimum and maximum ratios for the relevant blocks. I used the range of 0.45<ratio<0.6 for blue. For red I used anything greater that 0.7 or between 0.55 and 0.6 (to account for blocks on their side). I could have used a Matlab tool box for better image segmentation, and this would have given me more definite hue saturation values and region shapes, and thus these values would have been more accurate.