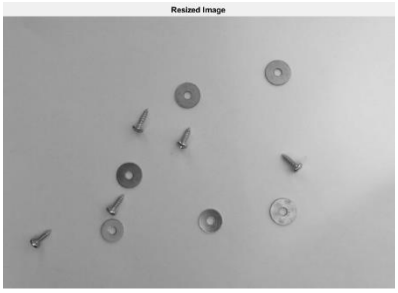
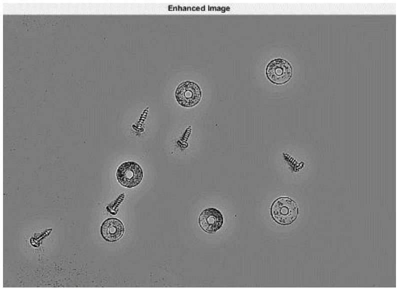
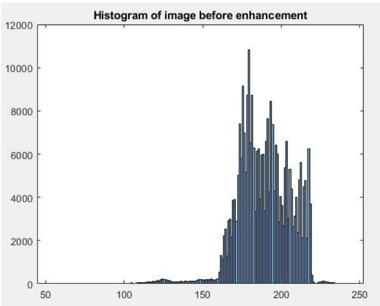
Image Processing Report

Task 1 – Pre-processing

The following images evidence the steps taken during pre-processing on the image ‘IMG_01.jpg’:

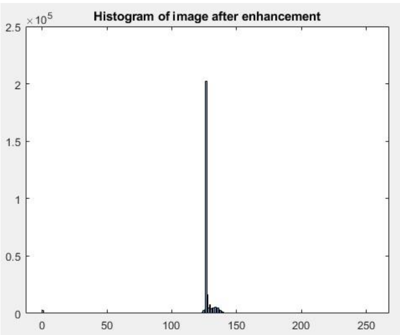






To generate the histograms and produce the images visually, the functions ‘histogram’ and ‘imshow’ were used. The image first underwent an adaptive histogram transformation (‘adapthisteq’) to produce multiple histograms of different regions within the image (Reza et Al, 2004). This conforms to spatial information within an image to increase edge contrast (Zimmerman et Al, 1988). The emphasis of this task is to improve edge contrast: details within each object are unimportant for object detection – only the edges are required. Sharpening is then applied to accentuate this point.

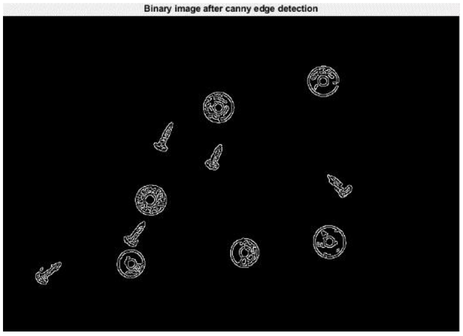
In an attempt to reduce irregular details such as accentuated sharpening around object shadows, a gaussian filter is applied to smooth over irregularities. After this stage, an image incomplement is generated and sharpened once more to highlight difficult areas within the image. This was then added back to the primary image to generate the mean of both images. A flat field correction was then applied to reduce nonuniform gain (Williams et Al, 2007), normalizing background lighting. Local contrast enhancement is then applied to ‘stretch out’ contrast in region of the image (Mukhopadhyay et Al, 2000). Finally, the image is sharpened one final time and then blurred using a gaussian filter.



After this stage, another histogram is produced to detail image changes numerically, and the binarized variant if formed from this. Otsu’s method is used (‘graythresh’ in Matlab) to find an optimum threshold for binarization. Otsu’s method iterates through all possible threshold values and attempts to find a minimum summation of foreground and background values (Bangare et Al, 2015). A final function ‘bwareaopen’ with a value of ‘9’ is used to remove pixels connected to nine other pixels or less.

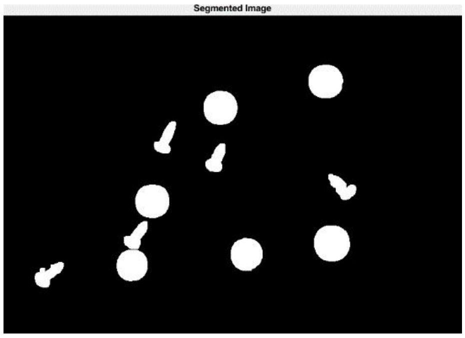
Task 2 – Edge Detection

Edge detection within this task is achieved using the ‘edge’ function, set to use canny edge detection. Canny edge detection is used for this task as it provides an effective response to unique edges, edge detection, and localization (Ziou et Al, 1988), through convolving an image with a filter and marking edges of output maxima. Two alternative detectors that could be used are Sobel and Prewitt, although they are unsuitable for this task. With the objects in the set of images being at least partially circular, Sobel and Prewitt are ill-advised as they are used to primarily detect vertical and horizontal edges (Shrivakshan et Al, 2012); information would be lost if they were to be used on circular objects.



Task 3 – Simple Segmentation

Edge detection combined with ‘imfill’ was not suitable for completely segmenting an image, as it is likely that edge results would not consist of complete shapes. To amend this issue, the image was dilated using two ‘strel’ functions, which extends the borders of each pixel by a desired amount. This increases the region of any object, potentially causing them to touch.

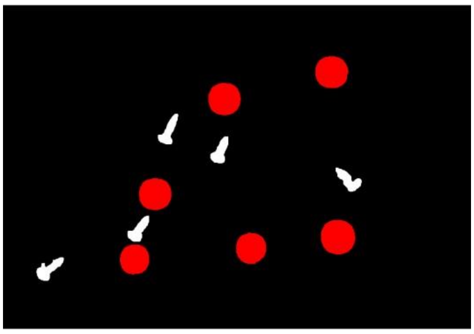


In order to segment any touching objects within the image, a watershed transformation is used to build a transformation mask and, using data gathered from ‘regionprops, divide objects by their centroids.

Task 4 – Object Recognition

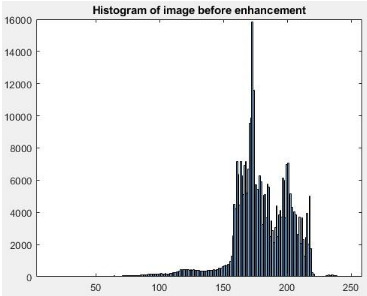
Within this assessment, there are three objects that must be determined from one-another: washers, short screws, and long screws. With these boundaries, it is easy to predict object classification based off the simple matter of the roundness of each object within the image itself. For example, the washers are close to perfect circles, thus any object with a high circularity is likely to be a washer (within reason). Similarly, short screws are less prismatic than longer screws, thus another margin between low and medium circularity can be drawn to identify these shapes.

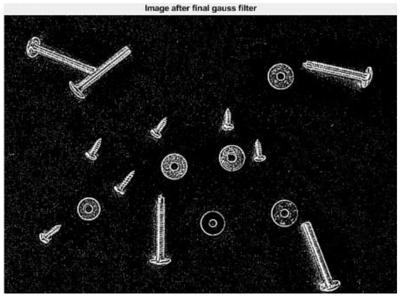
Despite the success of this method, it is potentially partial to failure if image enhancement is of a sub-par quality. Additionally, if more objects are added to identify (such as bolts) then this method of object recognition will have to be modified. This problem could be amended through adding additional methods of object recognition, such as general size comparative to other objects garnered through using ‘regionprops’.

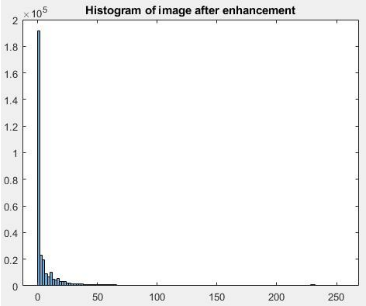


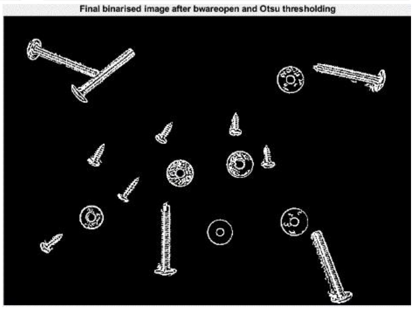
Task 5 – Robust Method

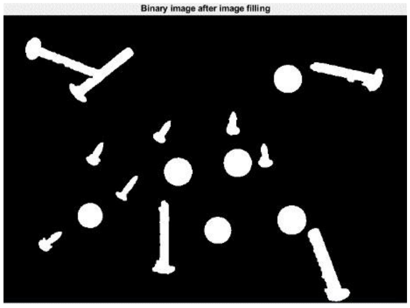
The following images evidence the steps taken during pre-processing on the image ‘IMG_05.jpg’:

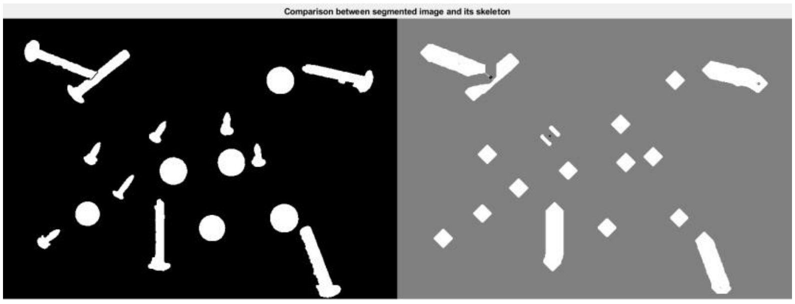


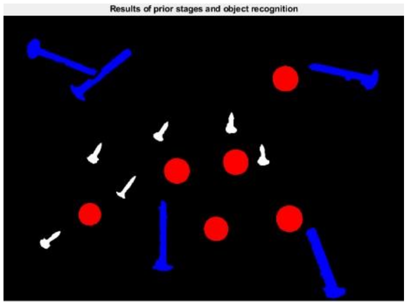












The robust stage changes much of the implementation within the stages of pre-processing and segmentation, although edge detection and object recognition remain unchanged.

Starting with changes to the pre-processing stage: the goal of the robust method is to normalise the background of each image whilst maintaining the integrity of objects. As shown in the third image within this section (the image after enhancement), a great deal of noise is generated from the enhancement methods taken.

These methods include the removal of adaptive histogram equalization, as this produced unsanitary results with excessive noise when combined with other methods. Instead, the function 'locallapfilt' is used, which acts on a similar basis as adaptive histogram equalization. Local Laplacian filtering works on the premise of building isotropic, spatially invariant, smooth Gaussian kernel (a Laplacian pyramid) within neighbourhood regions (Paris et Al, 2011). This creates an edge-aware filter that increases the dimension of pixels of a set size by any desired extent. Additionally, a bottom hat filter ('imbothat' using 'strel(disk, 15)').is used to fully segment object edges from the background. This process morphically opens and erodes the image to eliminate object shadow and background details, whilst the high strel size keeps objects similar. The amount of noise removed by 'bwareaopen' was also increased, to reduce the aforementioned increase of fallacious data.

Once more, edge detection is used as before. However, image segmentation differs to a great extent, as now pre-touching objects must be separated. This problem was mostly solved through manipulation and watershed of the enhanced image's skeleton. By default, touching objects will have a longer length than any singular object, so they can easily be detected. Thus, an image skeleton was generated by first eroding the image to remove any erroneous details ('imerode'), then the function 'bwmorph(*image*, 'skeleton', Inf)' was used. Short branches were reduced using 'spur'. Whilst 'bwskel' could have been used to prune the skeleton initially, 'spur' leaves a centroid on the skeleton, which is vital to the next stage of segmentation, where 'bwskel' does not.

After this, branchpoints were generated, and the skeleton was connected, thickened, then opened and thinned. The branchpoints were thickened and subtracted from the skeleton to identify splitting points for a watershed transformation, which was summarily applied to the actual image. If skeletons were fully removed, the watershed transformation would be prone to cutting through object regions, thus they remained to guide the transformation. After this, object recognition ran as stated earlier.

Yet despite this, the initial method of 'filling in the gaps' proved to damage results, as it forced objects to touch in some circumstance. To reduce the rate of 'false positives' in this regard, a 'bwmorph' function was used to modify the initial image before filling by using the 'close' parameter, which links open ends of an image. This created full objects of identical size to their prior.

Task 6 – Performance Evaluation

The table below summarises the success of common image segmentation methods:

	Dice Score	Precision	Recall	Accuracy	F-score	Similarity	Signal/Noise
Mean	0.50218	0.072	0.55184	0.94945	0.12665	0.91249	18.02007
Minimum	0.4714	0.0320	0.3410	0.9322	0.0586	0.8917	16.9766
Maximum	0.5614	0.1210	0.6368	0.9619	0.2020	0.9334	19.1260

It should be noted that the results generated from the linked code produce objects with a higher perpixel accuracy than those of the ground truth, meaning that all measures of success are likely to be somewhat diminished. Both recall and precision are of equal importance, but it is more likely that an image can be over-segmented/enhanced, leading to the lower precision. Of more importance is the dice score and similarity measures: with an average dice score of 0.5, this assumes that each image is somewhat similar to the ground truth. The similarity score suggests that both images are incredibly similar, with the minimum being 0.89. The peak signal/noise ratio indicates that there is little excess noise – such as shadows – within the output image. Overall, I believe that the images provided are of a high quality, with 9/10 correctly identifying all objects. Only one object is misinterpreted – that of a short screw misconstrued as a long screw in 'IMG_04.jpg'.

References

Bangare, S.L., Dubal, A., Bangare, P.S. and Patil, S.T., 2015. Reviewing otsu's method for image thresholding. *International Journal of Applied Engineering Research*, *10*(9), pp.21777-21783.

Mukhopadhyay, S. and Chanda, B., 2000. A multiscale morphological approach to local contrast enhancement. *Signal Processing*, *80*(4), pp.685-696.

Paris, S., Hasinoff, S.W. and Kautz, J., 2011. Local Laplacian filters: Edge-aware image processing with a Laplacian pyramid. *ACM Trans. Graph.*, *30*(4), p.68.

Reza, A.M., 2004. Realization of the contrast limited adaptive histogram equalization (CLAHE) for real-time image enhancement. *Journal of VLSI signal processing systems for signal, image and video technology*, *38*(1), pp.35-44.

Shrivakshan, G.T. and Chandrasekar, C., 2012. A comparison of various edge detection techniques used in image processing. *International Journal of Computer Science Issues (IJCSI)*, *9*(5), p.269.

Williams, T.C. and Shaddix, C.R., 2007. Simultaneous correction of flat field and nonlinearity response of intensified charge-coupled devices. *Review of Scientific Instruments*, *78*(12), p.123702.

Zimmerman, J.B., Pizer, S.M., Staab, E.V., Perry, J.R., McCartney, W. and Brenton, B.C., 1988. An evaluation of the effectiveness of adaptive histogram equalization for contrast enhancement. *IEEE Transactions on Medical Imaging*, *7*(4), pp.304-312.

Ziou, D. and Tabbone, S., 1998. Edge detection techniques-an overview. *Pattern Recognition and Image Analysis C/C of Raspoznavaniye Obrazov I Analiz Izobrazhenii*, *8*, pp.537-559.