CMP3108M Image Processing, Assessment Item One

O'Shea Douglas

1 Step 1

The imread matlab function is used to load the image.

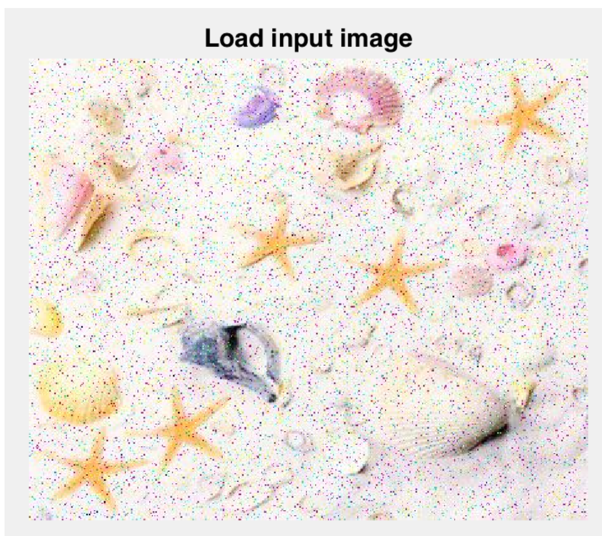

Figure 1: Loading Input Image

2 Step 2

In order to convert the image to grayscale, the function rgb2gray is used. The function works by eliminating saturation and hue information from the image whilst retaining its luminance. The image is now represented as a 2d array, with intensity values ranging between 0 and 255.

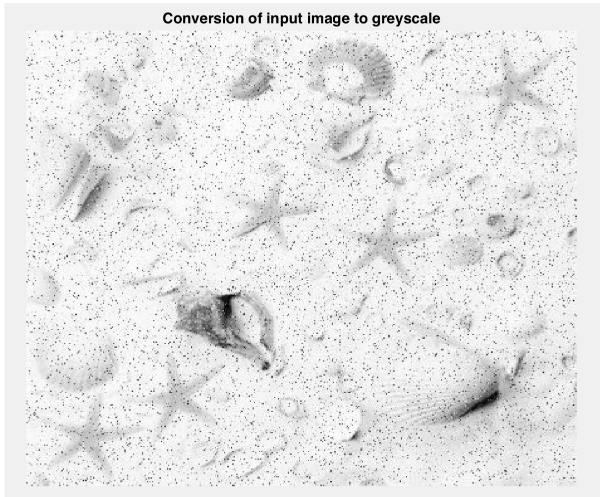

Figure 2: Conversion to greyscale

3 Step 3

3.1 Selected Method

A Median Filter has been used to remove salt and pepper noise within the image. As displayed in figure 1, the filter has worked well to remove the outlier pixels whilst retaining overall image quality. The filter determines each output value based on the pixels neighborhood. It computes the median value from the neighborhood list (including the input value), subsequently changing the input pixel with the median value.

3.2 Padding

Zero padding has been used. The median filter therefore has full coverage of the image including it corners and sides. Alternatively the image could be cropped. However this would reduce PPI losing image detail.

3.3 Filter Comparison

When testing a mean filter, output displays grey specks in similar positions to the salt and pepper noise. This is due to the linear nature of the filter being more sensitive to the 0 and 255 outliers (Quote). Pixels true to the original image are being influenced by outlier pixels in their neighborhood. With its trait of selecting the middle value within a pixels neighborhood a median filter is less sensitive to these outliers and therefore in this case a better choice for noise removal than the mean filter.

A Gaussian filter seems to remove the salt and pepper noise however does not preserve edges of the image as well as the median filter. In terms of speed the linear trait of the Gaussian algorithm is likely to be faster than median filter's non-linear sorting process. However maintenance of detail is crucial in later stages i.e. identifying the starfishes. As a result detail over speed is an accepted trade off.

A Wiener filter has been used to determine if the tailoring of local image variance removes more noise than a median filter. It seems when defining a 3 by 3 neighborhood, the filter doesnt blur the edges of objects within the image however it has only removed a small amount of salt and pepper noise.

In evaluating a pixel using a 5 by 5 neighborhood noise is taken away, however this is at the cost of losing further detain within the image. As displayed in figure 1, a 3 by 3 neighborhood completely takes away the salt and pepper noise whilst also retaining image detail. If there was a higher amount of noise within the image, using a 5 by 5 neighborhood may have been applicable.

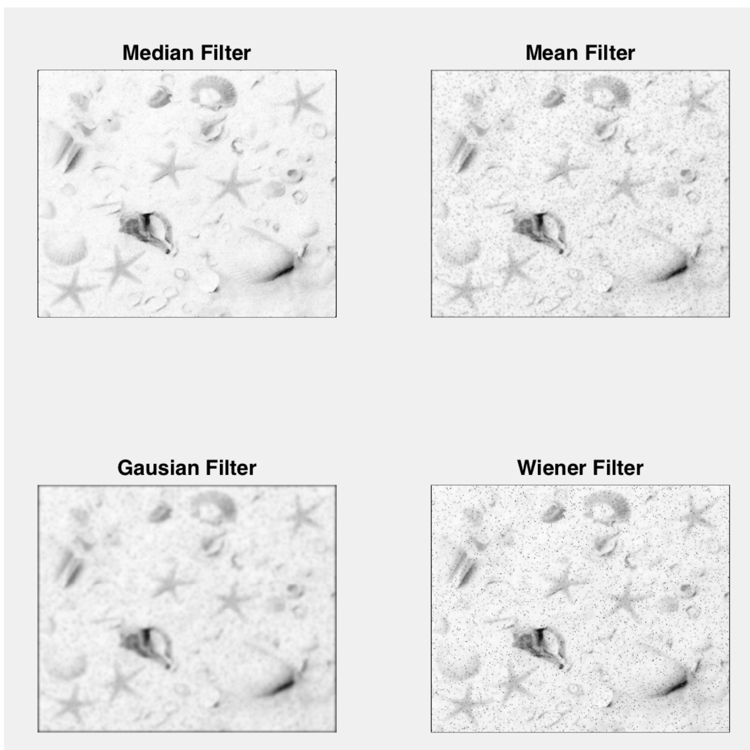

Figure 3: Filtering

4 Step 4

4.1 Selected Method

Histogram equalisation is used to enhance the contrast of the image. The method outputs the images intensity levels by approximation of their probability of occurrence. As displayed in figure 1, the original image has histogram components, which are localised to high intensity values. As shown in the HE algorithm, the probability of the starfishs intensity is high. This makes output of each starfish darker, which aids in his or her segmentation from the rest of the image. However there is a lot of detail around the starfishes with a similar shade. This has led to areas around the starfish have becoming darker in contrast therefore increasing unwanted detail around each starfish.

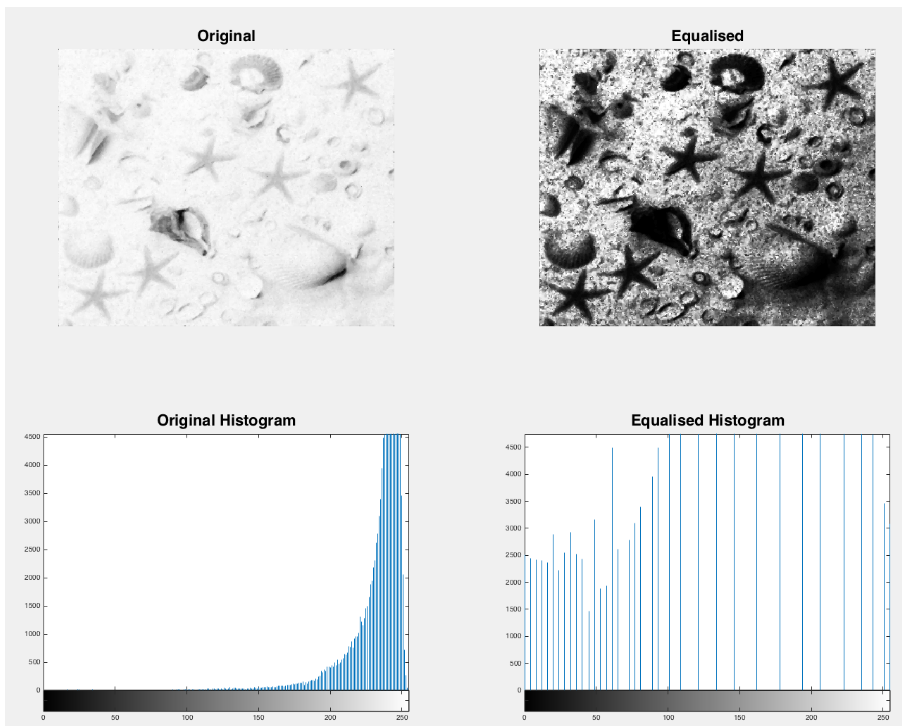

Figure 4: Histogram Equalisation

4.2 Enhancement Comparison

Original intensity levels of the image are narrowly localised around the higher end of the intensity spectrum. Contrast stretching has subtly expanded the narrow range making the pixel intensities slightly darker. This has slightly increased the darkness of the starfish. However, contrast stretching has stretched already dark pixel intensities of large (non star-fish) artefacts to an even darker end of range, which makes them more defined than each starfish. From this, Histogram equalisation appears to be a better option for enhancement. This is because the high probability of each starfishs intensity leads to a darker output. As a result, it returns more definition to each starfish than contrast stretching where edges are more defined.

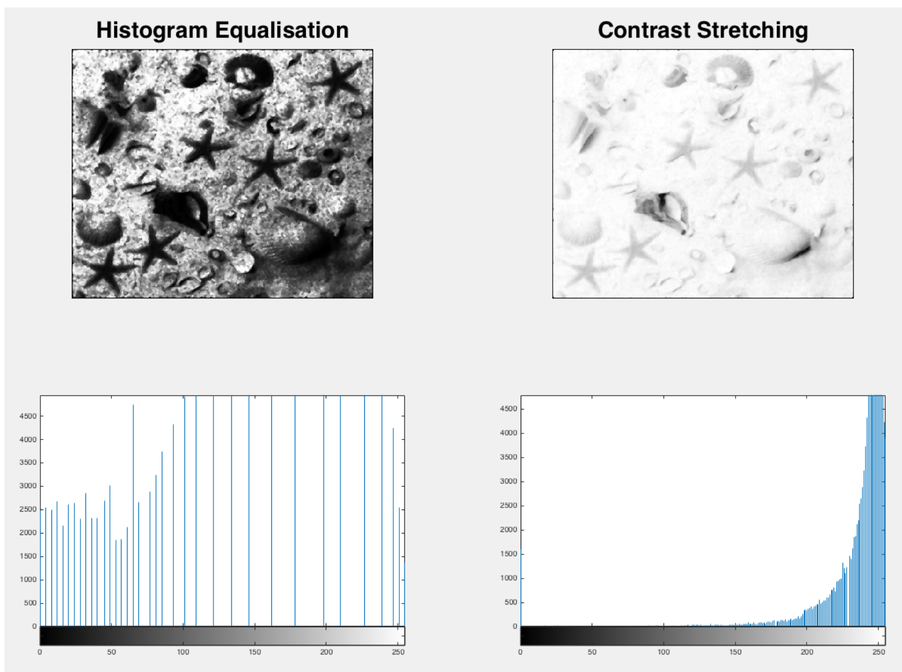

Figure 5: Histogram Equalisation

Gamma correction has also been tested as an enhancement. Gamma correction works using the power law where an input value is raised to a given power and then outputted as a non-linear value. When specifying a power of 0.5 (Gamma compression), brighter intensity levels increased. This did not increase the contrast of the starfishes, but does make background detail lighter, which would be useful in its filtering when thresholding. When specifying a power of 1.5, darker intensity levels increased. This inversely increased the contrast of the starfishes but also made background details darker. This particular result was similar to the output of histogram equalisation. Histogram equalisation however seems like the best enhancement to use, this because it makes the edges of each starfish darker which therefore defines them more than any form of gamma correction.

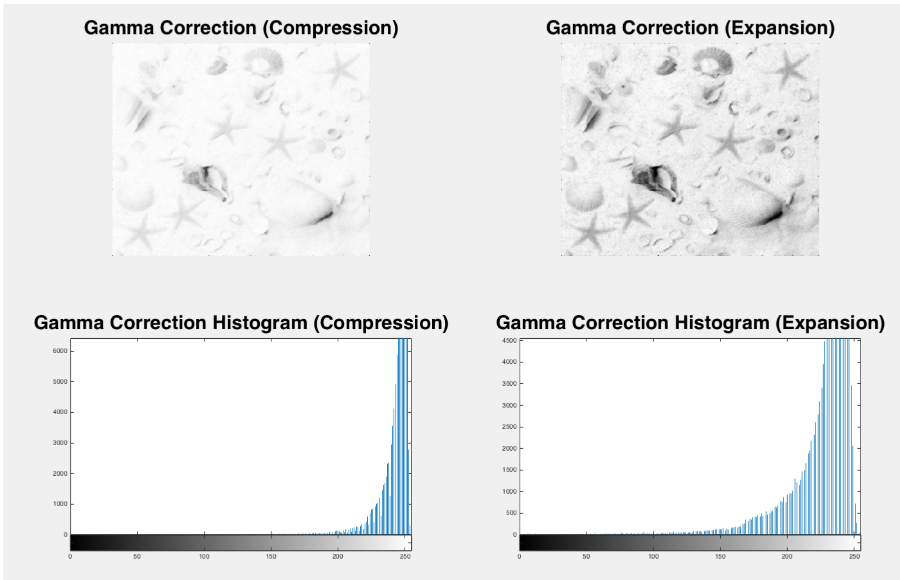

Figure 6: Histogram Equalisation

5 Step 5

The Otsu method has been considered. The method works by splitting the image intensity values into two regions. The variance of both regions is then calculated which indicates the compactness of both distributions. Adding both variance result returns the threshold for the image.

Because the image has been equalised pixel intensities are not group together and not bi model in histogram structure. As a result the threshold seems too high and enables undesirable detail as foreground pixels. This has resulted in white details forming around or connecting to the starfishes, as well as to other non-relevant shells.

Instead of the Otsu method a simple threshold number has been trialled and tested. It seems to be the best value, which segments the pixels of interest from the background.

The in built Matlab function im2Bw is used to segment the image into classes. It works by evaluating every pixel within the image against a threshold t. If a given pixel value is above t then the value is mark as '1' (forground) else the pixel is marked as 0 (background).

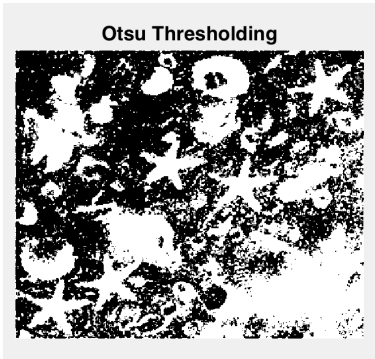

Figure 7: Otsu Threshold

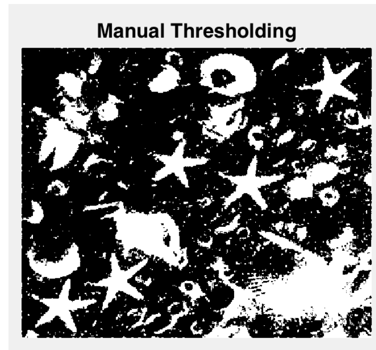

Figure 8: Manual Threshold

6 Step 6

6.1 Selected Method

The high intensity probability of background pixels within the image leads to histogram equalisation increasing the detail of background artefacts, which do not belong to the pixels of interest.

Morphological processing inhibits a set of non-linear operations, which can be used to remove the imperfect artefacts within the binary image. Opening has been utilised. Opening comprises of two operations, erosion and dilation. Both operations create a new binary image where a structuring element Kernel determines whether a pixel has a zero or non-zero value. Each morphological kernel has a pixel arrangement, which specifies the evaluative region of the input image. For example the square shaped kernel in figure 8 will consider a complete 3 by 3 neighbour of a pixel, whilst the kernel in figure 9 will only consider its corresponding pixels plus shaped.

6.2 Erosion Comparison

Erosion superimposes the structuring element se on top of the binary image where the origin of se is placed at every computed pixel in the input image. If the structuring elements meet non-zero neighbourhood pixel from evaluated the given pixel stays the same, otherwise the pixel is converted to 0. A disk shaped kernel with a size 5 x 5 (2 pixels each side of the evaluating pixel) is used to erode the image. As displayed in figure 10 this eliminates small background artefacts whilst still retaining general shape of each starfish. When using a when using a square shaped kernel, unwanted shapes in the image are reduced to a smaller size however edges of the stars seem to lose a higher amount of detail, this is because of the square structure evaluates more pixels than the disk arrangement than the disk arrangement.

In using a larger size 7 x 7 disk kernel more unwanted detail is removed from

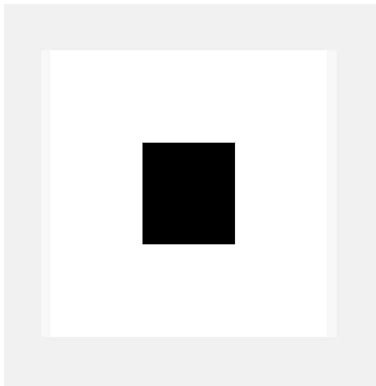

Figure 9: 3x3 Square Kernal

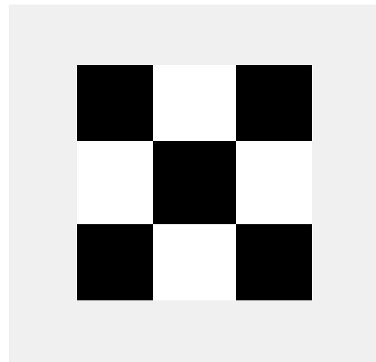

Figure 10: 3x3 Disk Kernal

the binary image. However this is at the cost of further reducing the shape of each starfish (figure). The subsequent process of dilation would recover this shape, however a higher kernel size would also be required. This would increase the size of other artefacts as a well as the pixels of interest. As a result a 5 x 5 disk size seems like the best compromise between removal of artefacts and maintenance of starfish shape.

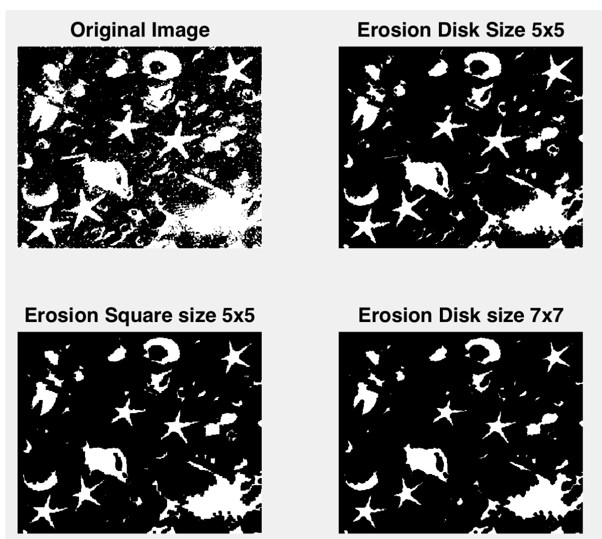

Figure 11: Dilation Comparison

6.3 Dilation Comparison

Dilation works in an opposite manner to the erosion process. If one or more pixels in the structuring process correspond with a pixel in the binary images neighbourhood, the evaluated pixel is changes to 1, otherwise the pixel is already one and another pixel is evaluated. Using a square kernel arrangement with a size of 4 x 4 increases the size of each starfish without compromising shape. When using a disk arrangement with a size of higher 5x5, the shape of each starfish do not enlarge as much as using a 4 x 4 square arrangement. This is due to the shape of the disk-structuring element, which does not evaluate all surrounding neighbourhood pixels. In using a 5 x 5 kernel size with a square kernel, the starfishes bloat whilst increasing the size of the unwanted artefacts. Whilst using a smaller size 3 x 3 kernel, the starfishes have sharp edges, which are otherwise smoothed out when using a 4 x 4 size.

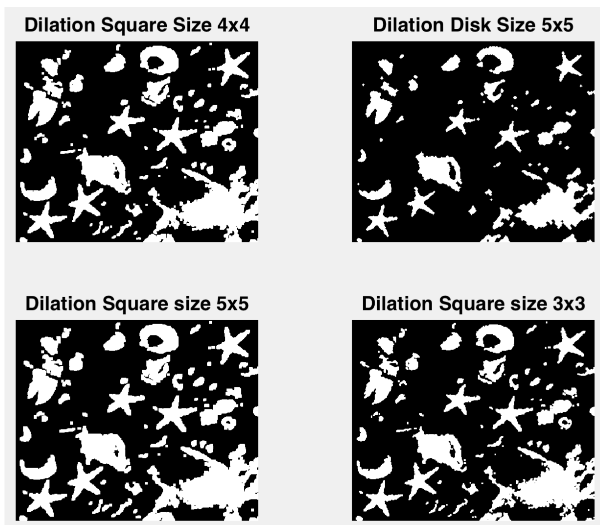

Figure 12: Dilation Comparison

When implementing a closing operation (dilation and erosion), this maintains more wanted artefacts than the chosen opening method. This is a result of the small background pixels being enlarged first, then not being taken away due to the erosive kernel not being large enough to remove the artefacts. A size kernel which removes the unwanted pixels also deforms the pixels of interest (starfish). As a result the closing operation is not utilised in the binary image.

References

[1] Bernhard E. Riecke; Jacob B. Freiberg; Timofey Y. Grechkin. (2015). Can walking motions improve visually induced rotational self-motion illusions in

virtual reality? . Journal Of Vision. 15 (3), 3.

[2] Katy Vans. (2015). Playing mindgames in a neuroscience, art and tech vision of the future. Available: https://www.theguardian.com/technology/blog/2015/jun/18/playing-mindgames-in-a-neuroscience-art-and-tech-vision-of-the-future. Last accessed 20th Oct 2016.

[3] Luca Turchet. (2015). Designing presence for real locomotion in immersive virtual environments: an affordance-based experiential approach. Virtual Reality. 19 (1), 277-290.

[4] Grechkin, T.; Azmandian, M.; Bolas, M.; Suma, E. (2015). Towards context-sensitive reorientation for real walking in virtual reality. Virtual Reality (VR). - (-), 185 - 186.

[5] Arsalan Heydariana, Joao P. Carneiroa, David Gerbera, Burcin Becerik-Gerbera, Timothy Hayesc, Wendy Wood. (2015). Immersive virtual environments versus physical built environments: A benchmarking study for building design and user-built environment explorations. Automation in Construction. 54 (-), 116-126.

[6] Sharif Razzaque, Zachariah Kohn, Mary C. Whitton . (2001). Redirected Walking. Eurographics 2001 - Short Presentations. 1 (-), -.

7 Appendix

Below are the complete EA1 and EA2 forms.