1 Part 1: Thresholding

1.1 Introduction

Thresholding is a technique used to segment an image based on grey level intensities within the image. Two common methods of thresholding an image are applying a "Fixed Global Threshold" and applying an "Adaptive Threshold". Each of these methods take a greyscale input image, and output a binary (black and white) image.

1.2 Techniques

Fixed Global thresholding involves applying a single threshold value across the image, i.e. if the intensity value of the pixel is greater than the threshold value, set that pixel to white, otherwise, set it to black.

Adaptive thresholding techniques base their threshold values at the current pixel off the neighbouring pixels. The "5x5 Adaptive Thresholding" method utilised in this section takes the mean intensity value of the 24 pixels surrounding the currently selected pixel as the threshold value for the currently selected pixel. If this center pixels intensity value is greater than the threshold, it's value is set to white, otherwise it is set to black.

1.3 Pseudocode

1.3.1 Part a

1. Load the image into Matlab using the "imread()" function
2. Use the "rgb2gray()" function to convert the image to greyscale.
3. Display the greyscale image using the "imshow()" function.

1.3.2 Part b

1. Follow the steps of Part a.
2. Apply a fixed global threshold using the VSG package 'Threshold' function. An arbitrary value should be chosen for the fixed threshold value.
3. Vary the fixed threshold value in order to obtain optimal background/foreground segmentation, while retaining facial features.
4. Display the thresholded images using the "imshow()" function.

1.3.3 Part c

1. Follow the steps of Part a.
2. Apply a 5x5 adaptive threshold using the VSG package "5x5Thresh" function.
3. Display the adaptive thresholded image using the “imshow()” function.

1.3.4 Part d

1. Follow the steps of Part a.
2. Apply Gaussian noise to the greyscale image using the MIP “imnoise()” function. Zero-mean noise should be used for this section.
3. Execute the fixed global thresholding and 5x5 adaptive thresholding procedures as described in part b and part c respectively.
4. Adjust the variance value of the “imnoise()” function. Repeat steps 2 and 3.

1.4 Results

1.5 Conclusion

2 Part 2: Segmentation

2.1 Introduction

2.2 Techniques

2.3 Pseudocode

2.3.1 Part a

2.3.2 Part b

2.4 Results

2.5 Conclusion

3 Part 3: Convolution

3.1 Introduction

3.2 Techniques

3.3 Pseudocode

3.4 Results

3.4.1 Part a

3.4.2 Part b

3.4.3 Part c

3.5 Conclusion

4 Appendix