# Laplacian Edge Detection

This method of edge detection can uses the difference between a pixel and the surrounding pixels to test for edges. This is done using a kernel, where a larger kernel can be used to reduce noise more. The most common kernel used is the 3x3, and is the one used in this program. The kernel is slid from pixel to pixel so that all pixels except the outer edge are given a new value. The value assigned is 8 times the central pixel minus the sum of the 8 surrounding pixels. This can be represented graphically:

-1 -1 -1

-1 +8 -1

-1 -1 -1

This algorithm makes edges appear lighter and non-edges appear darker. This algorithm is only effective on images that have first been greyscaled however because of the way colour is stored as RGB in images.

# Smoothing

Smoothing takes the mean of all the pixels in a kernel in order to smooth out edges in images. To do this a kernel is used, and in this program it is a 3x3 kernel. It is slid from pixel to pixel so that all except the outer edge of pixels have their value changed to that of the mean of the 9 pixels in the kernel. The mean is calculated as:

Total of all pixel values / Number of pixels (in this case 9)

# Noise Removal

This algorithm takes the median of all the pixels in a kernel in order to remove noise whilst smoothing the image minimally. To do this a kernel is used, and in this program it is a 3x3 kernel. It is slid from pixel to pixel so that all except the outer edge of pixels have their value changed to that of the median of the 9 pixels in the kernel. The median is calculated as:

When all values in the kernel are placed in order of magnitude, the middle value is the median.

In practice in this program this means sorting the 9 elements and then choosing the 5th element as the middle value.

# Red Eye Reduction

To reduce the so called “red eye” effect in photographs the program must look at each pixel in turn and decide whether it is red or not. To decide this a “tolerance” is used, where a pixel with the red value above some threshold but the green and blue values below some other threshold is considered red. This only works where pixels are stored in RGB(α) format.

If a pixel is deemed red it will have its colour changed to a colour similar that to that of normal eye; in this program a 50% grey is used. Non-grey pixels retain their original colour.

# Greyscale (Extension)

Greyscale is used to change a colour image into a grey image. In this instance this is to improve the effect of the Laplacian Edge Detection algorithm. To change an RGB (Red, Green, Blue) value to gray each colour value must be extracted then multiplied by a fraction based on how important each colour is to human vision. A good approximation, which is used in this program, is: 0.3\*R , 0.6\*G, 0.1\*B. From this you can tell that green is the most important colour for the human eye, and that blue is the least important. These multiplied values are then added together and this is the grey value. the final step is to set all three colour channels of the pixel to this grey value.