Digital and Embedded Systems
Professor Thomas Bräunl

Tutorial 9 – Image Processing

1. Implement an RGB to Hue conversion in C.

- Read an input camera image in RGB format
- Convert every pixel to a HSI values.
- For a given desire hue value and if I (intensity) is above a threshold, generate a binary truth image, where a pixel is true if its hue is within a fixed range of the desired hue

Example:

2. Implement the Laplace and Sobel edge detectors as functions in C.

- Image input and output parameters should be of type BYTE *
- Make sure that the output value will stay in range 0..255, e.g. by limiting the max (min values)
- Use local neighbor pixels to calculate new value for each pixel
- Do not use sqrt or other compute-intense functions. Use sum ob absolute values instead.

Laplace 3x3:

0	-1	0
-1	4	-1
0	-1	0

Sobel 3x3 (x-direction and y-direction

-1	0	1
-2	0	2
-1	0	1

1	2	1
0	0	0
-1	-2	-1

Note: Sobel = $| Sob_x | + | Sob_y |$

3. Implement a thinning operator for a binary image as a function in C.

3x3 Kernels:

0	0	0
	1	
1	1	1

	0	0
1	1	0
	1	

If a pixel matches exactly the given 3x3 kernel, then it is deleted (set to black); otherwise it stays unchanged.

For every pixel, you need to apply all four 90°-rotation variations of each of the two kernels, so 8 kernels altogether are applied to each pixel.

The output for each pixel is either 0 or 1.

Example: