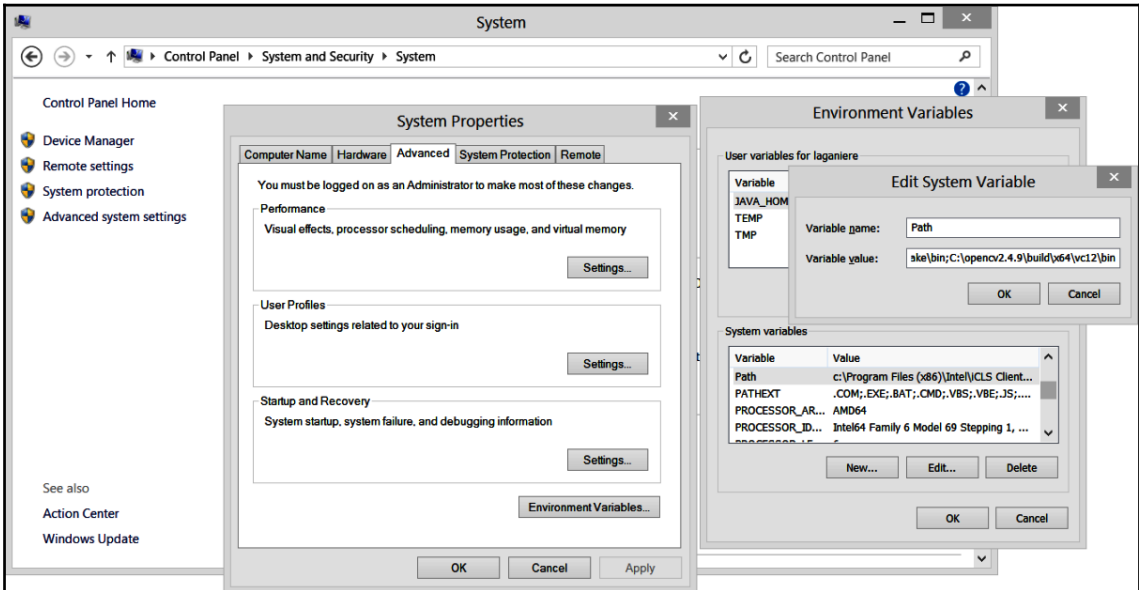
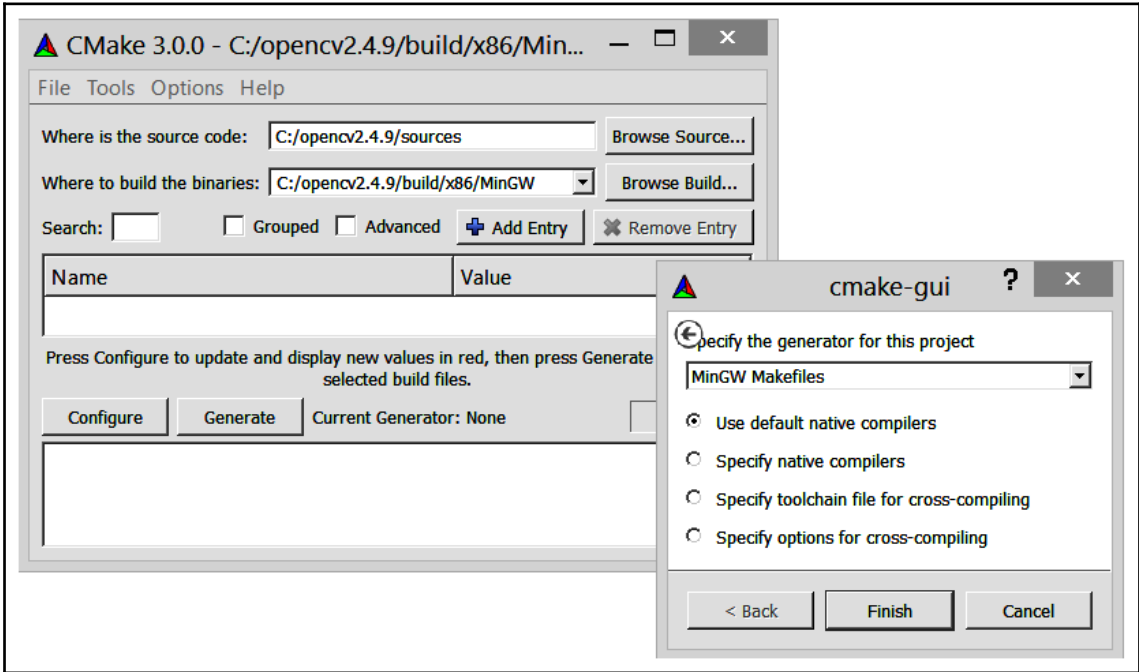
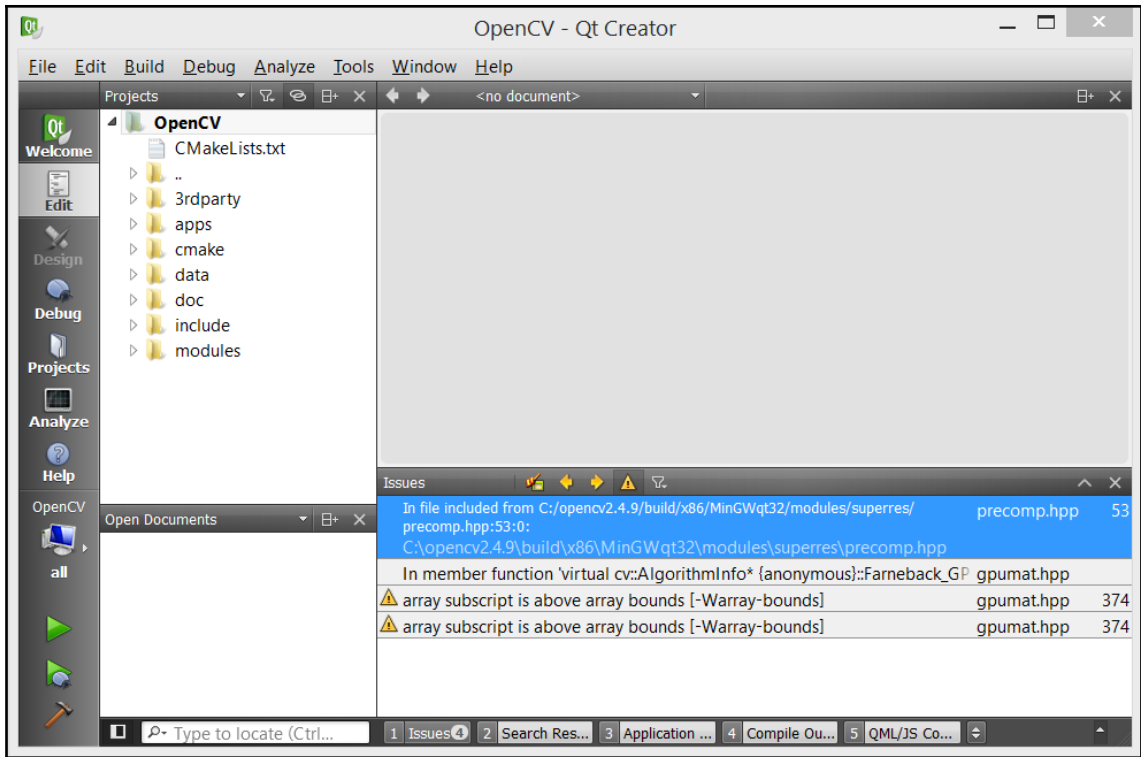
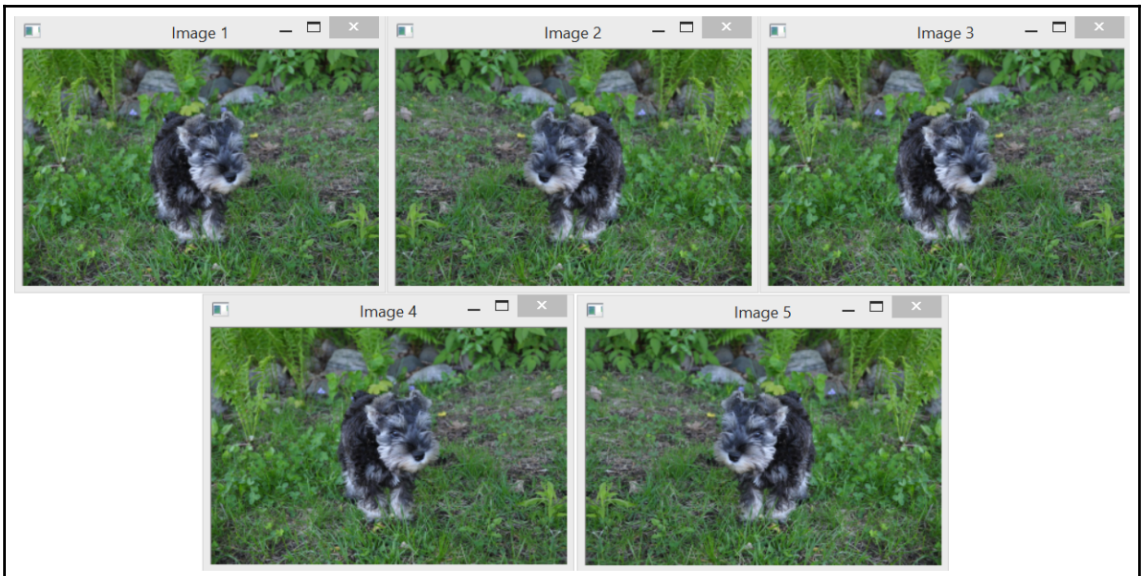


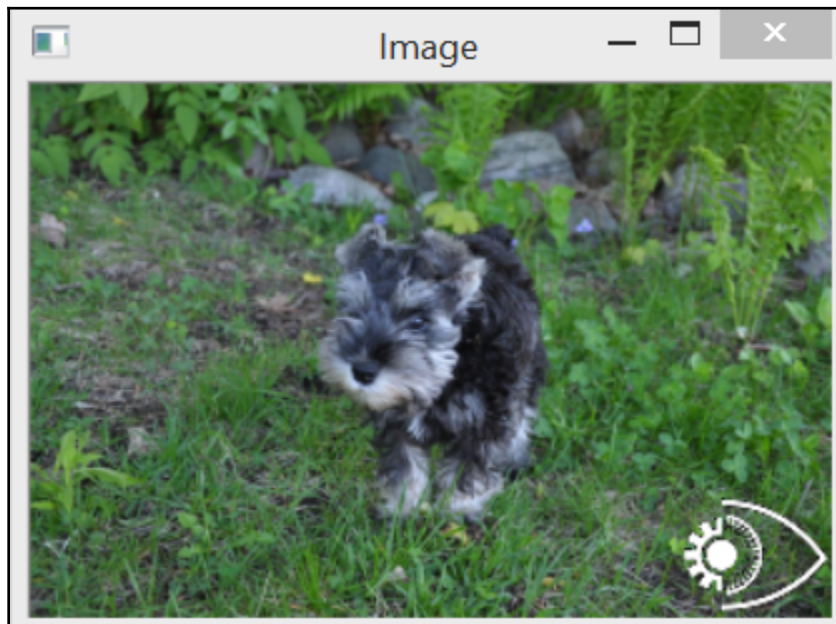
Chapter 1: Playing with Images



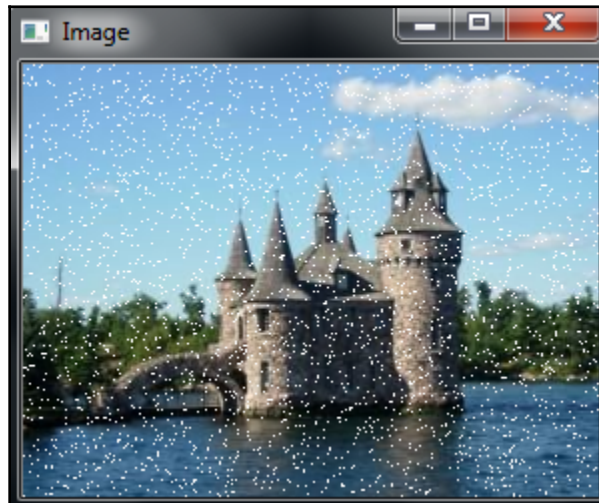


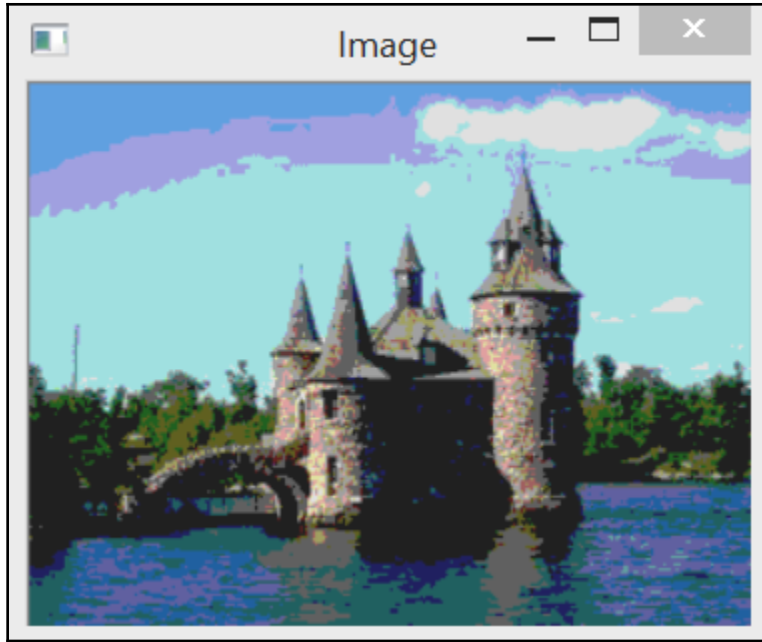






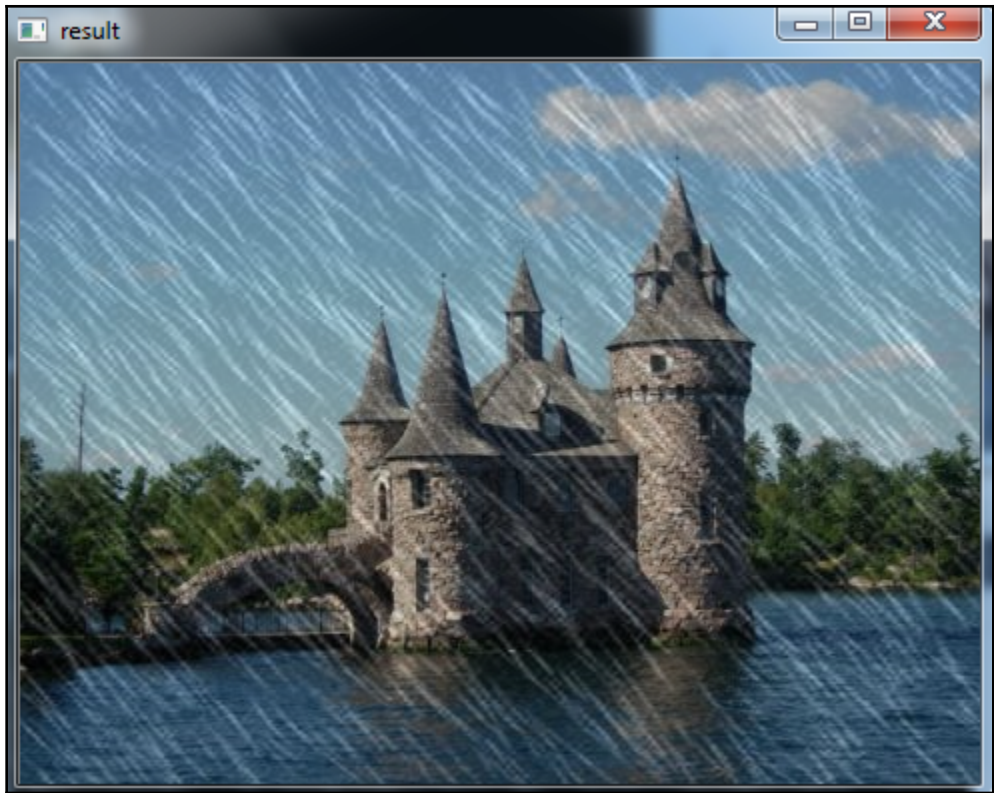
Chapter 2: Manipulating the Pixels

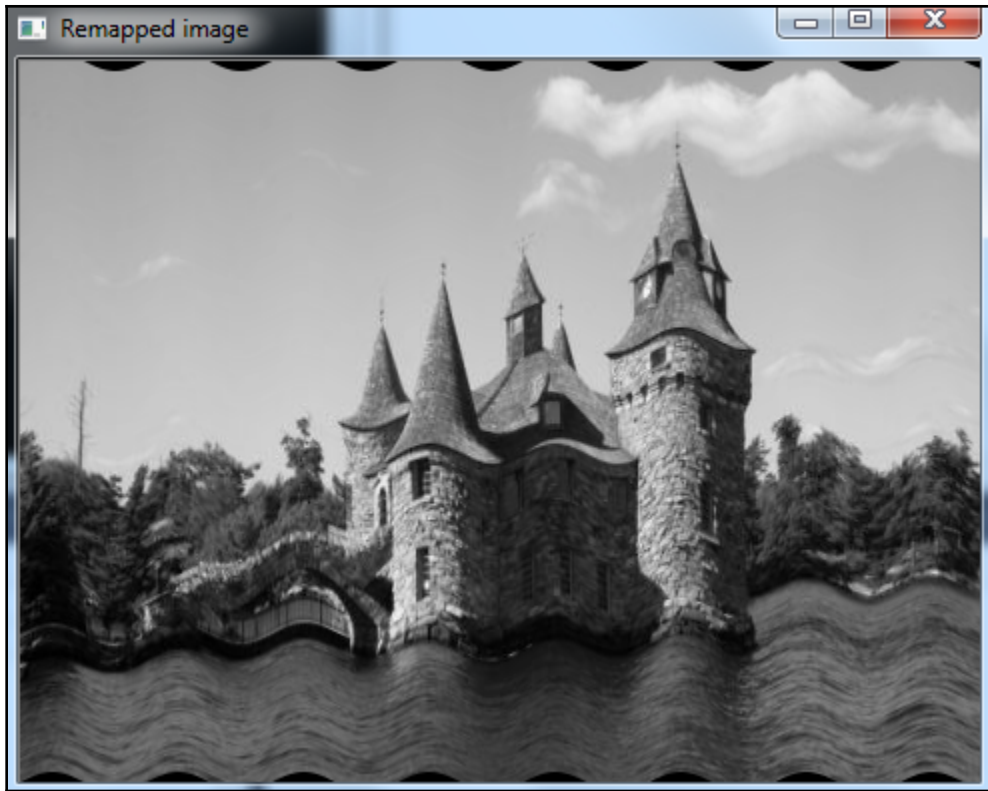




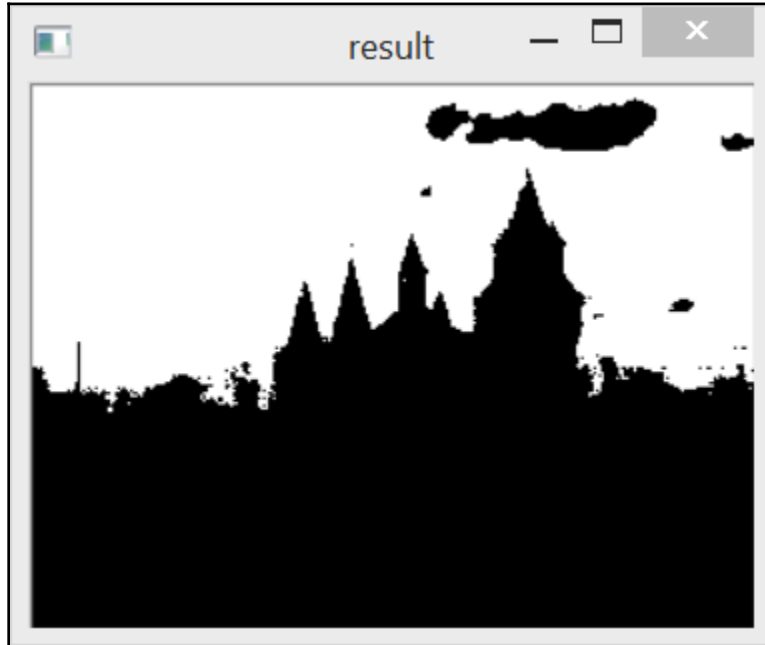


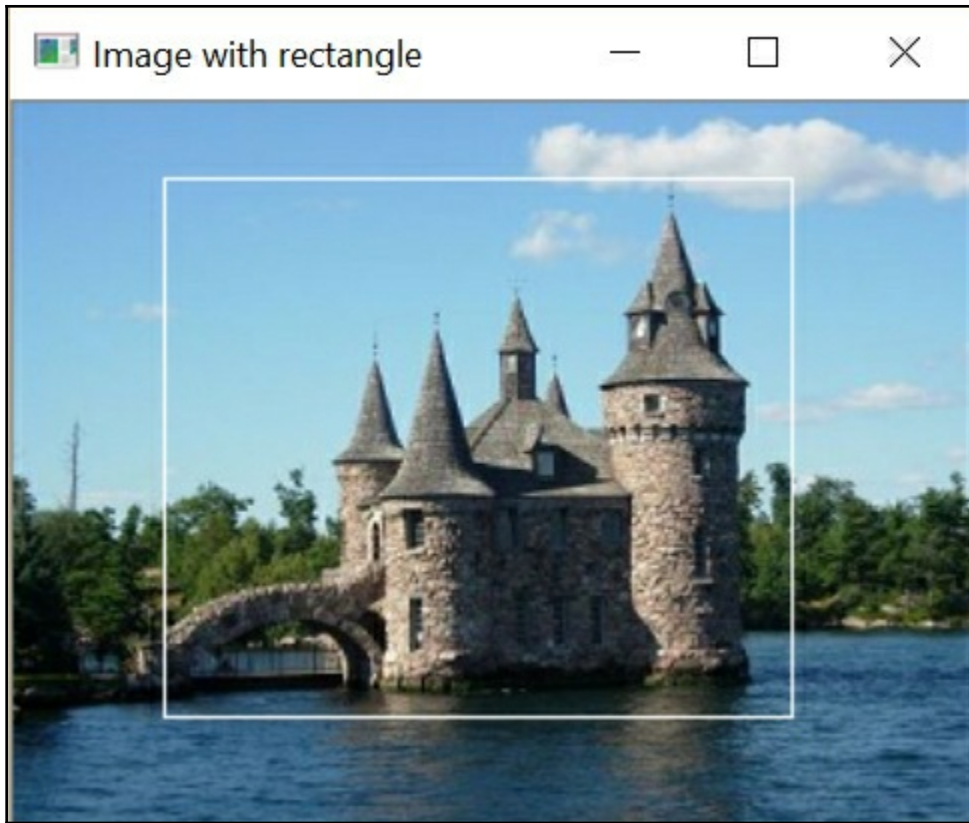


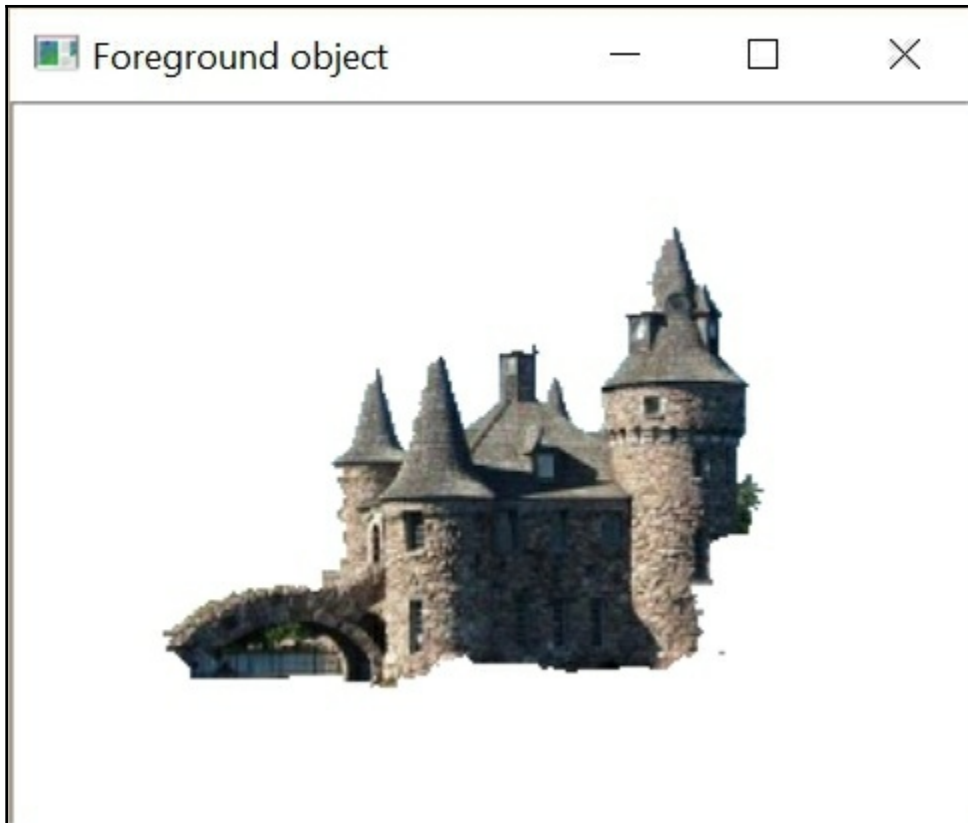


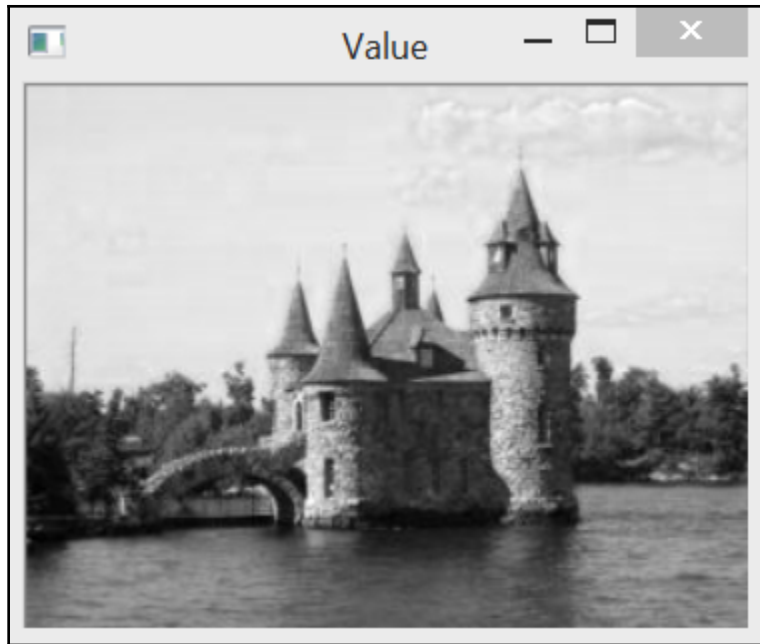


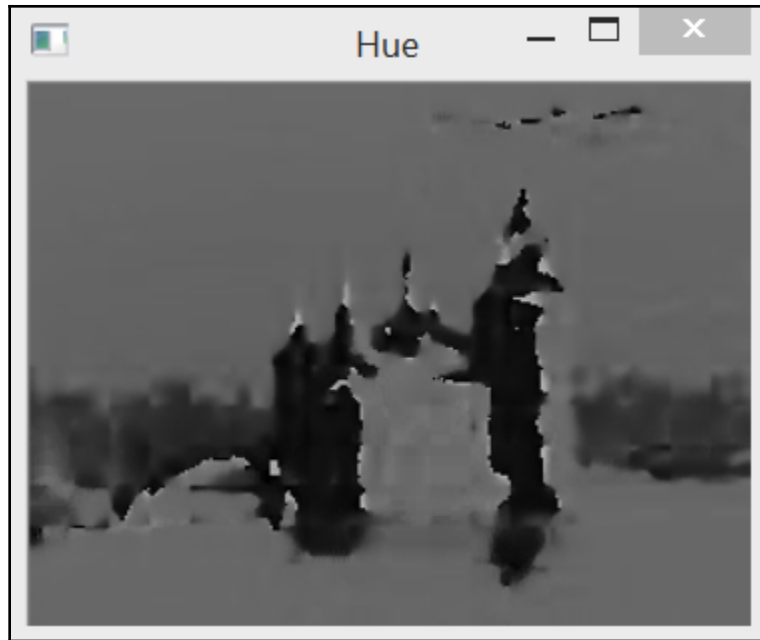
Chapter 3: Processing Color Images with Classes

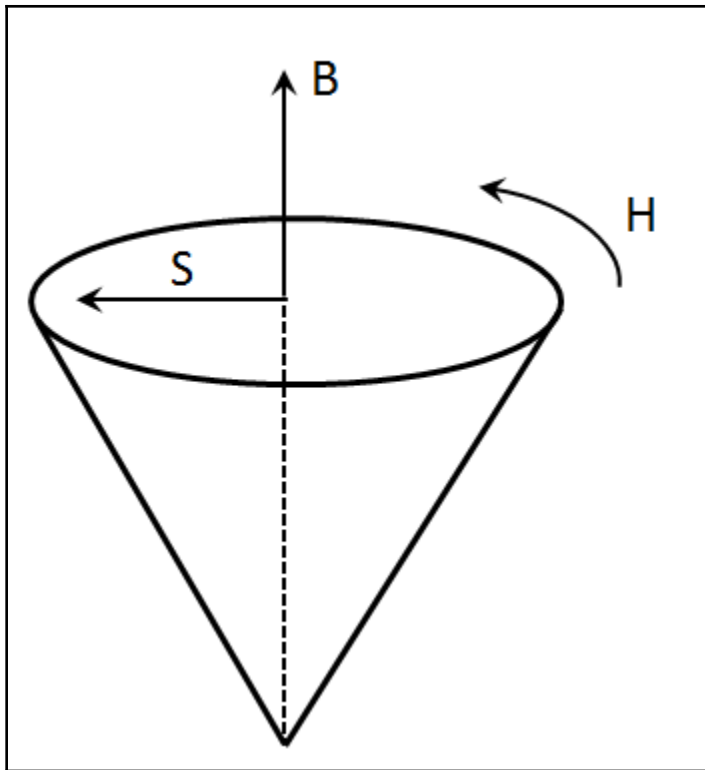


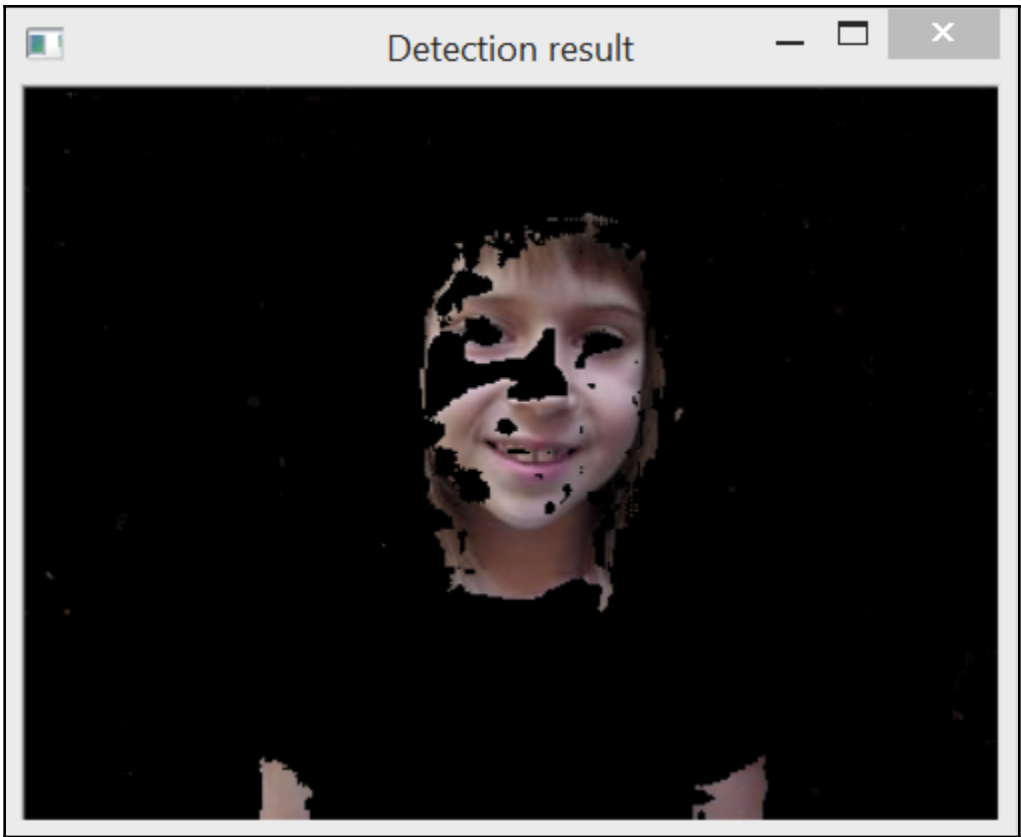




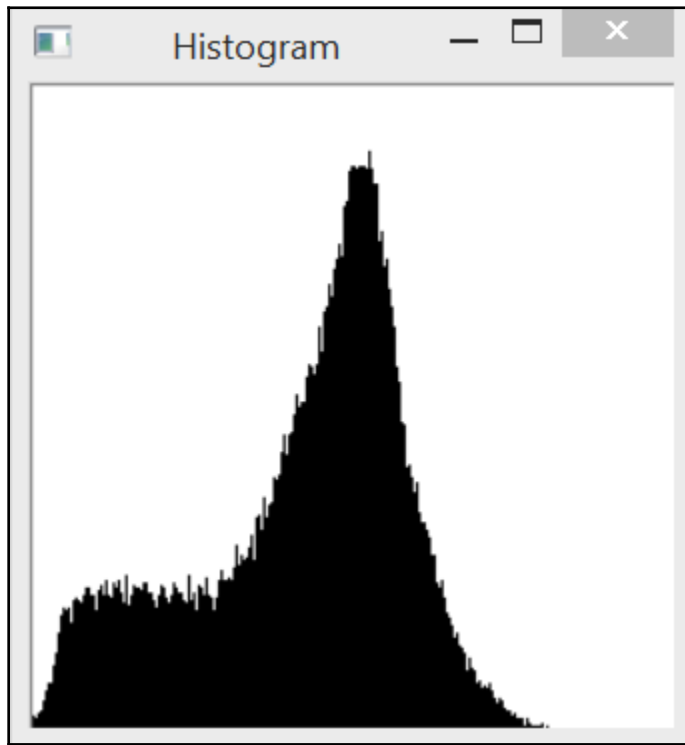


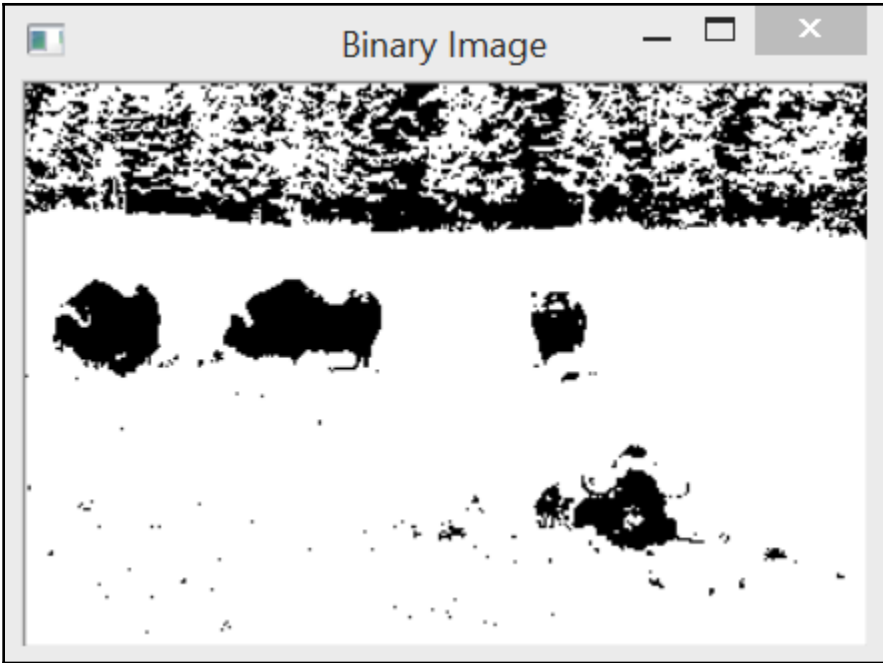




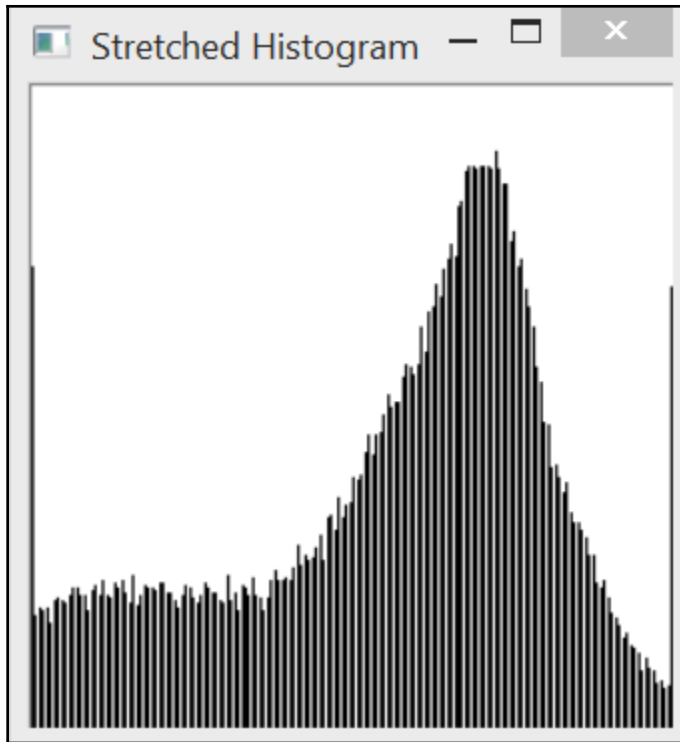


Chapter 4: Counting the Pixels with Histograms

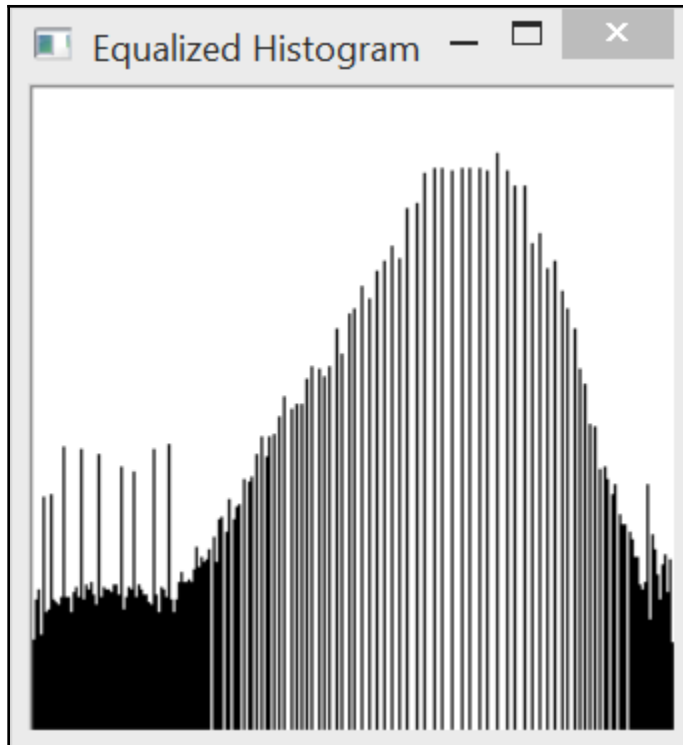


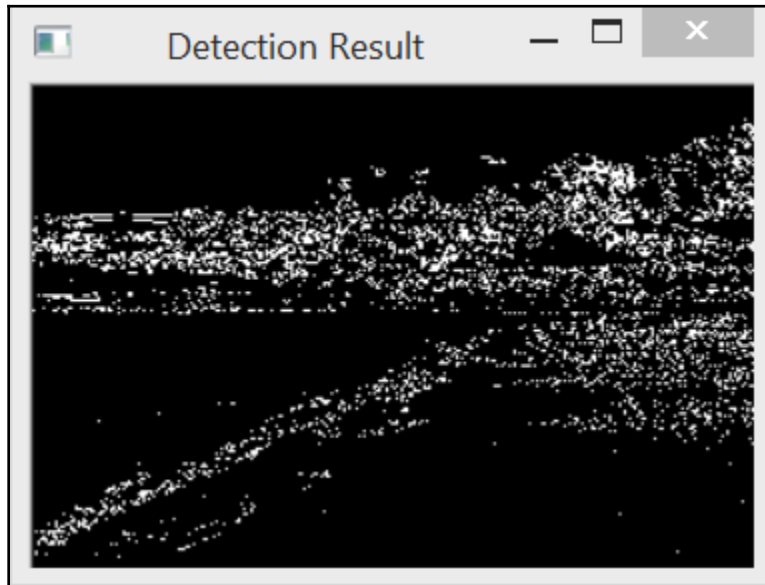
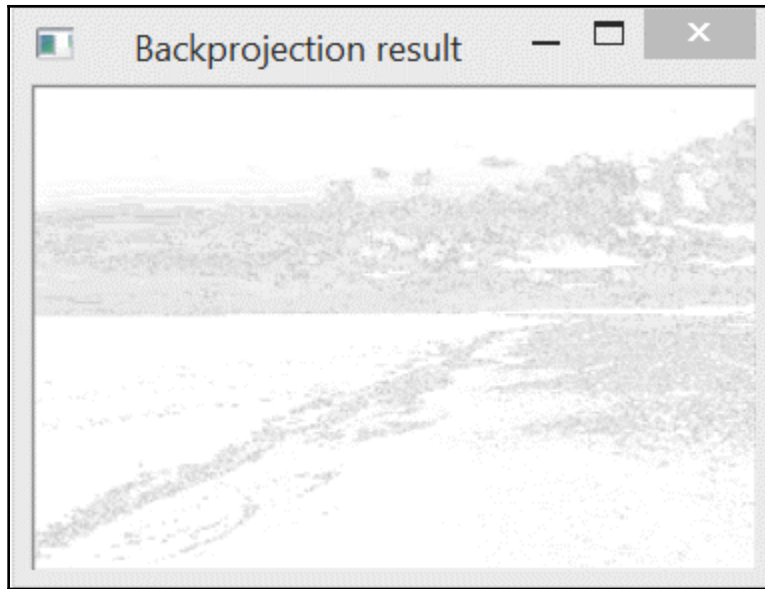


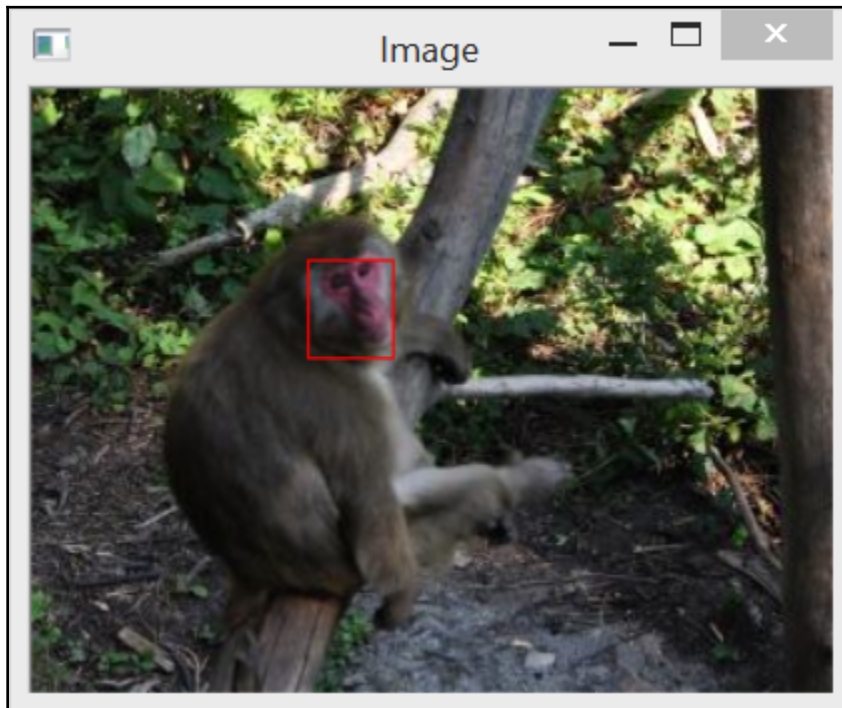
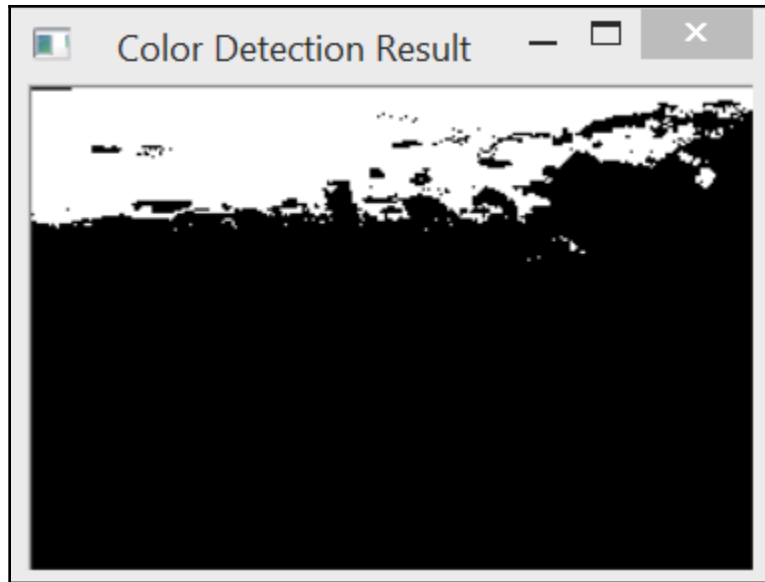


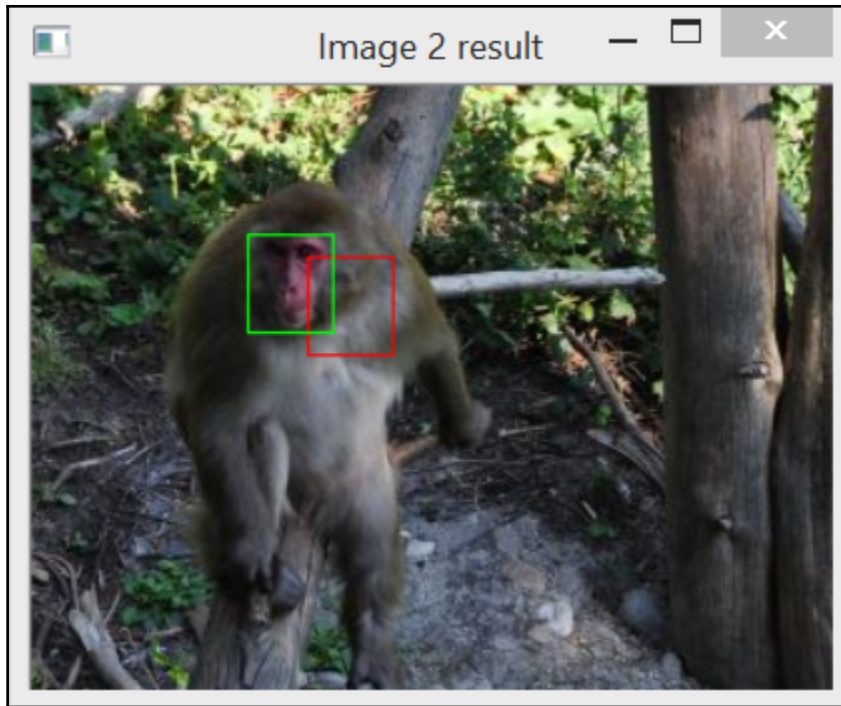




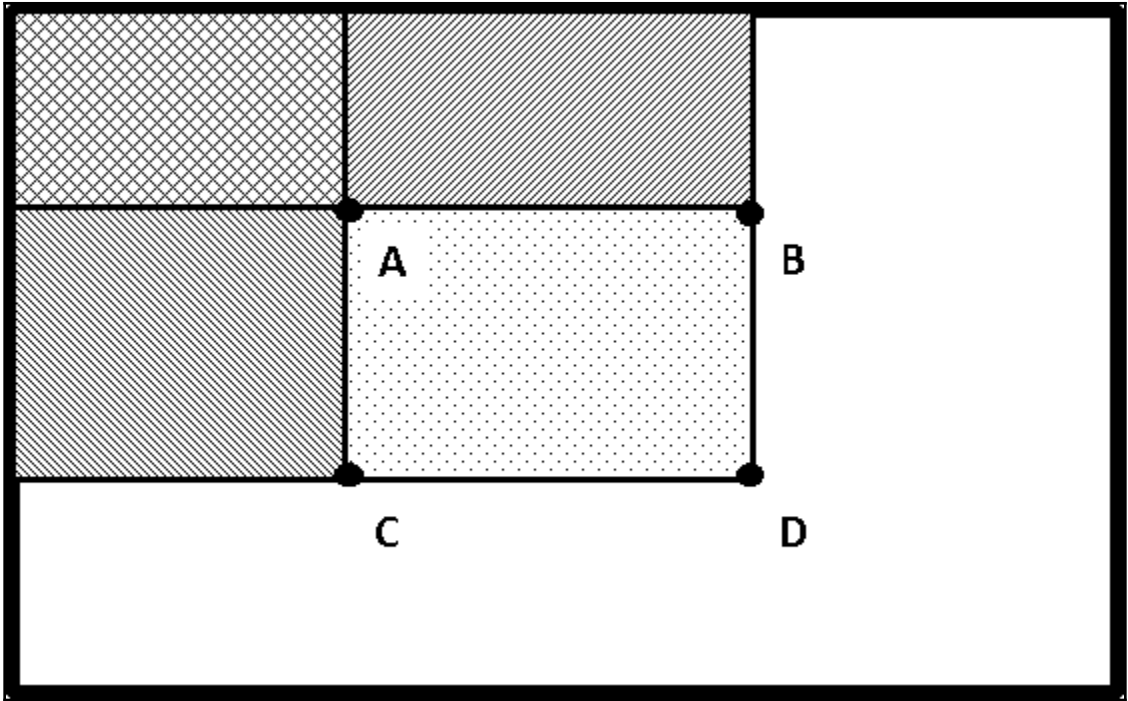


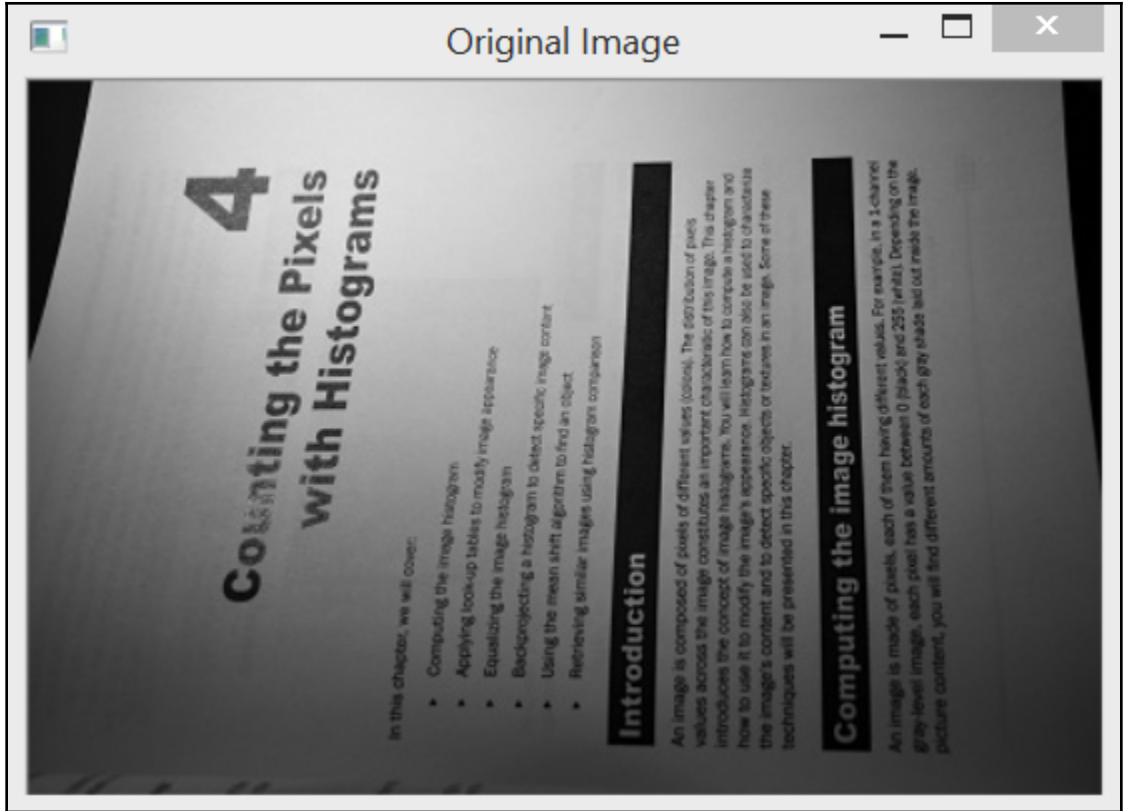


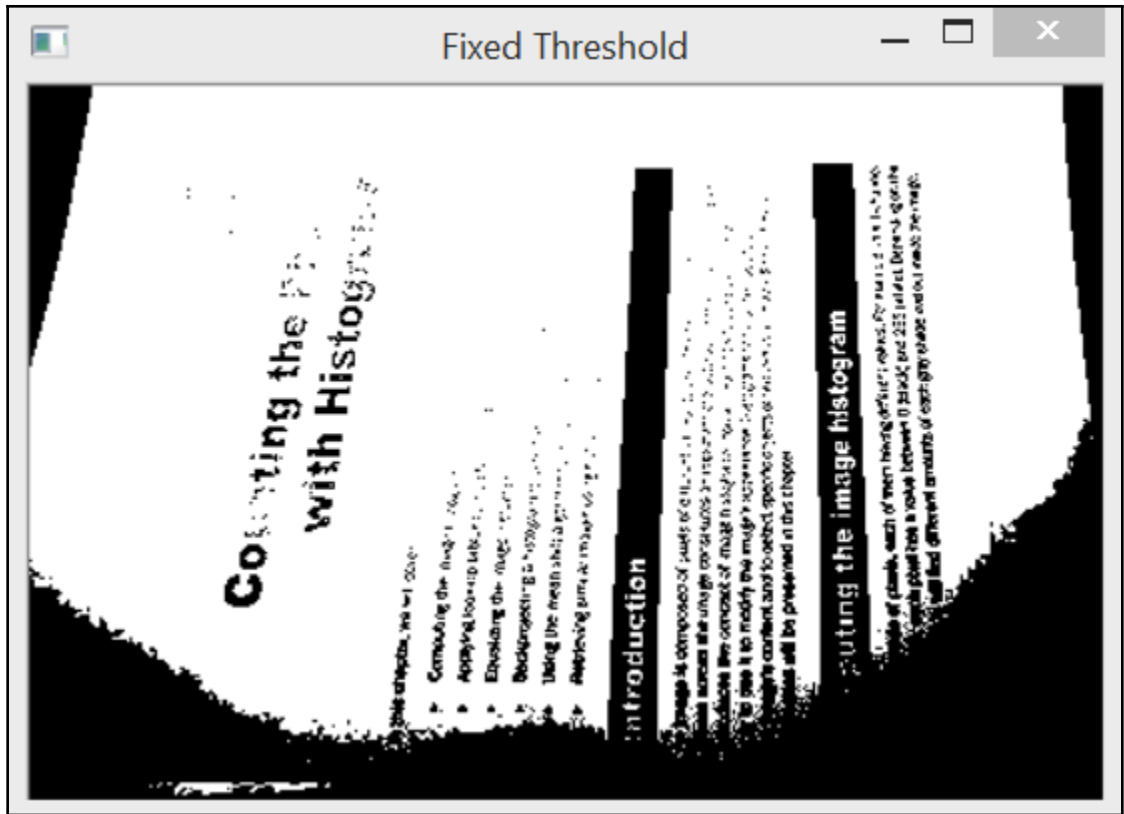












4

Counting the Pixels with Histograms

In this chapter, we will cover:

- Computing the image histogram
- Applying look-up tables to many image histograms
- Equalizing the image histogram
- Designing a histogram to detect specific image content
- Using the region split algorithm to find an object
- Recognizing similar images using histogram comparison

Introduction

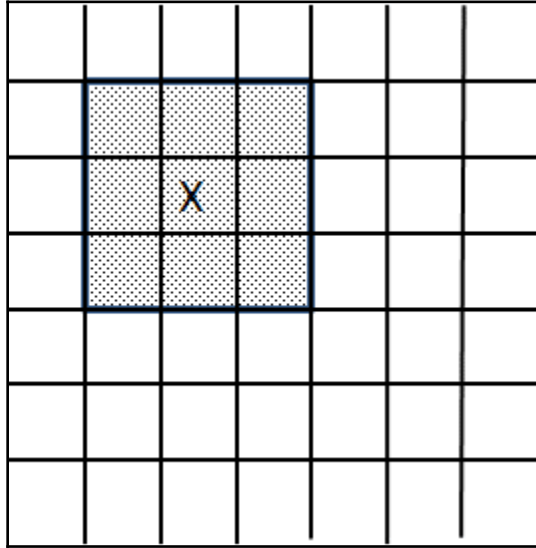
An image is composed of pixels of different values (color). The colors have a level value across the image considered as important variations of features. This chapter introduces the concept of image histograms. You will learn how to compute histogram and how to use it to modify the image's appearance. Histograms can also be used to characterize the image's content and to detect specific objects or features in an image. Some of these techniques will be presented in this chapter.

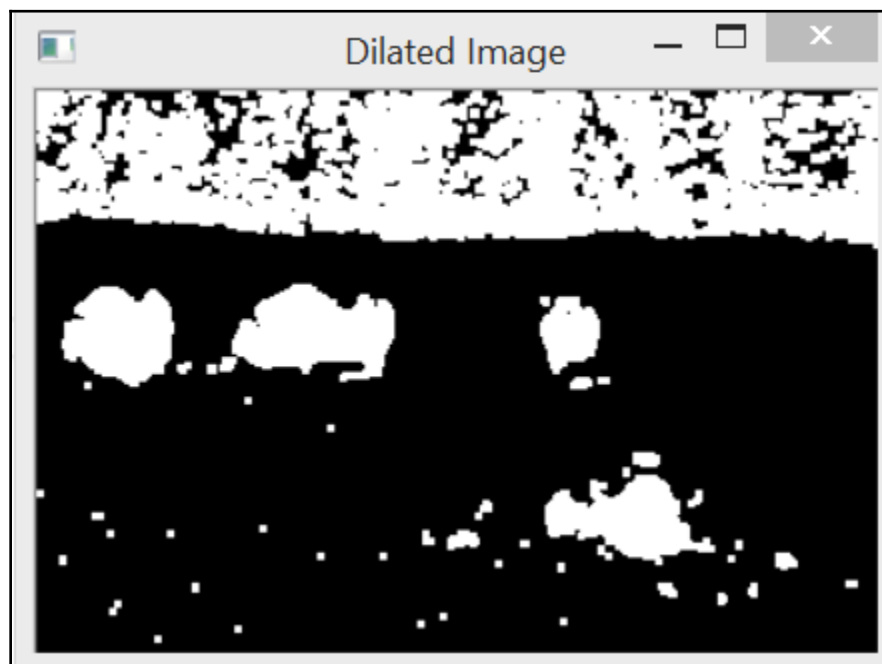
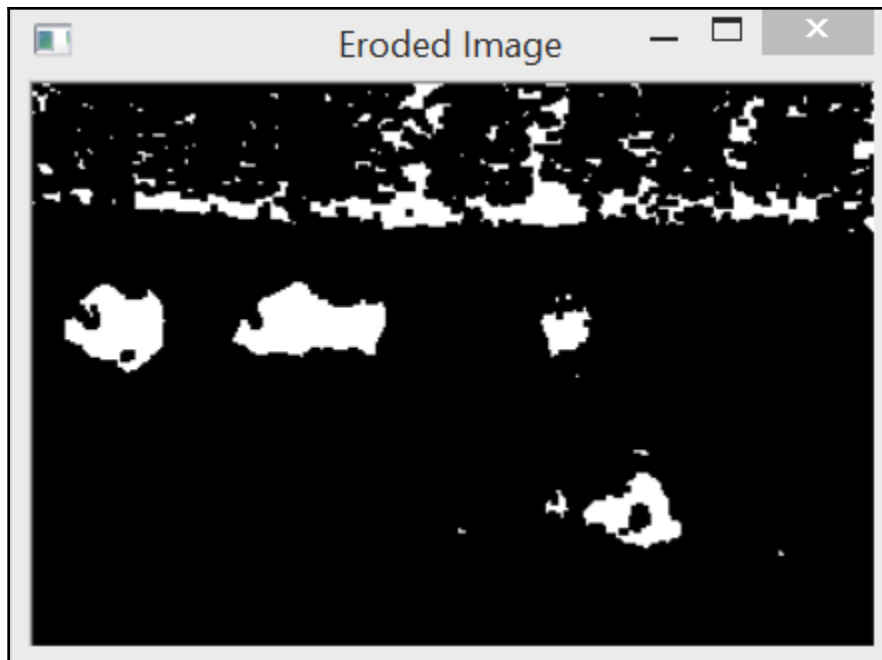
Computing the image histogram

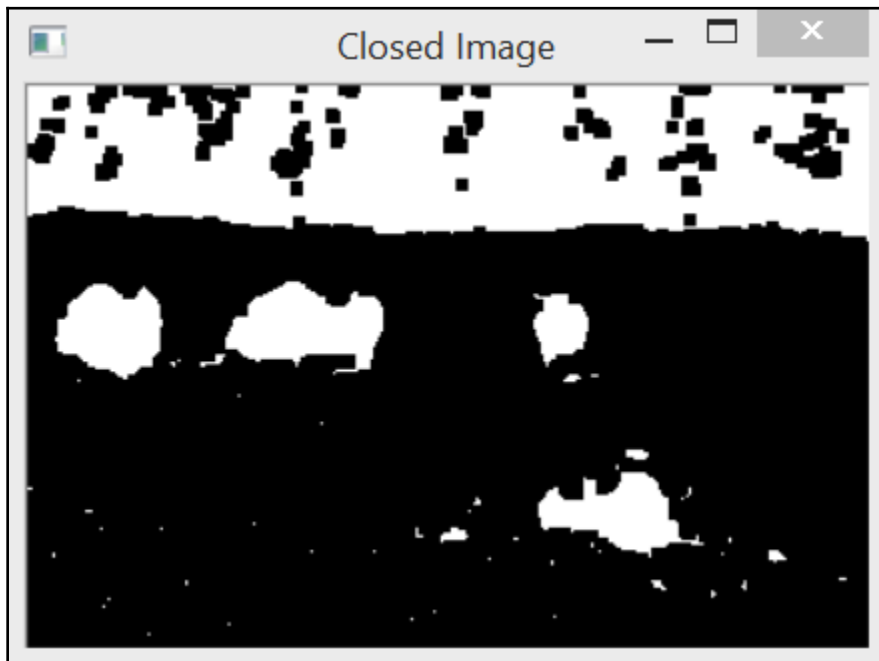
An image is made of pixels, each of them having a given value. For example, in a 2-channel gray-level image, each pixel has a value between 0 (black) and 255 (white). Depending on the picture content, you will find different amounts of each gray-level pixel and, through

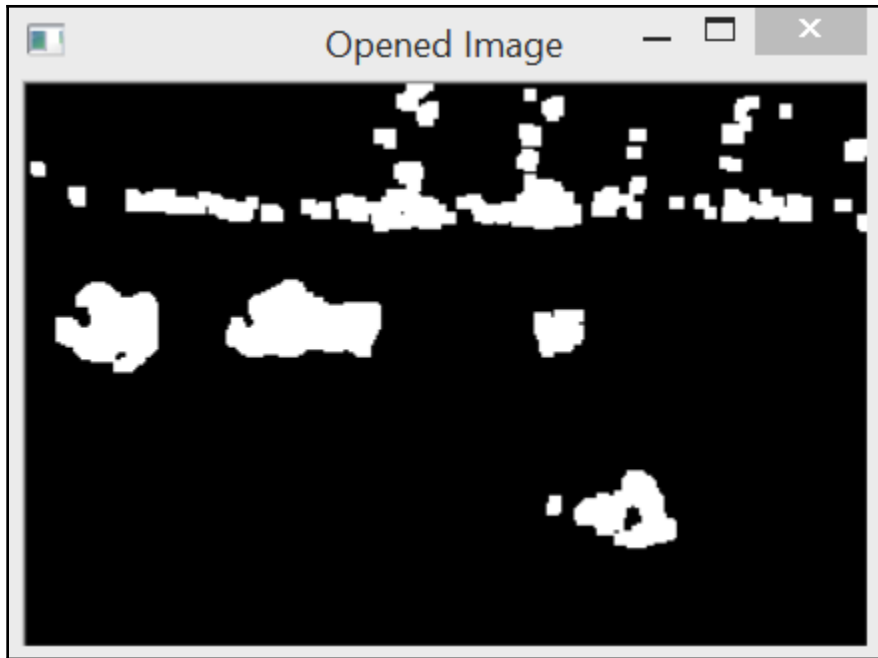


Chapter 5: Transforming Images with Morphological Operations



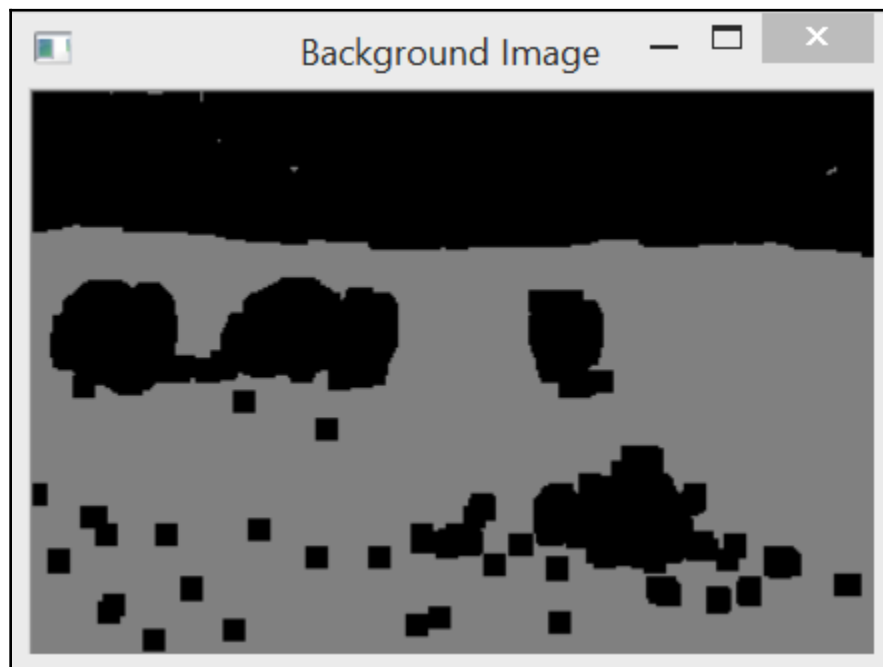
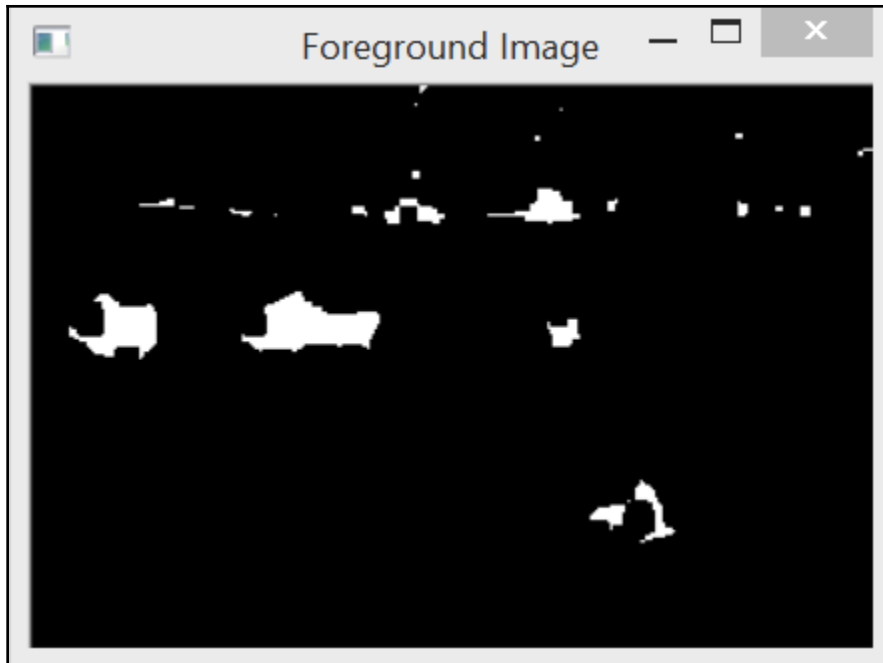


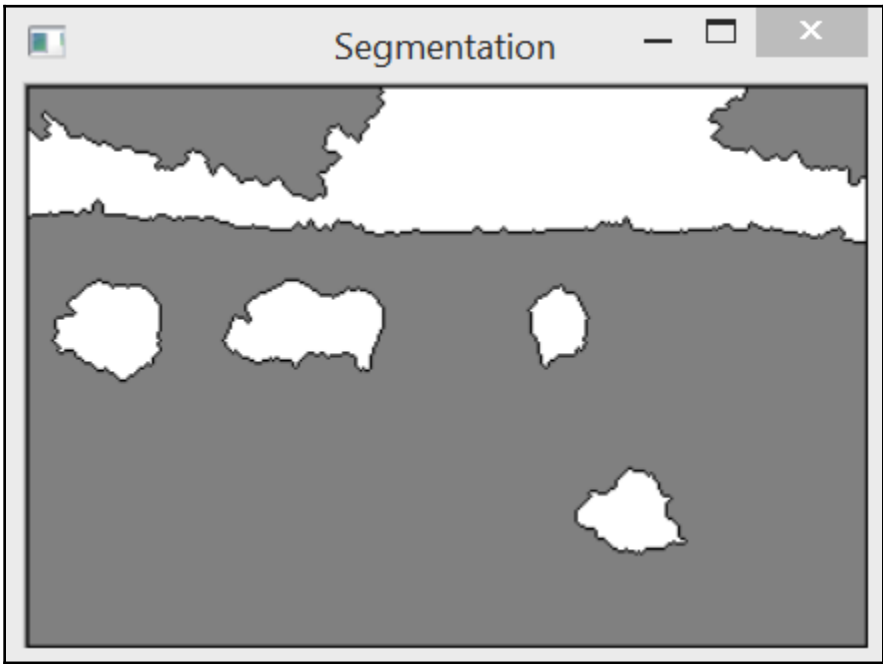
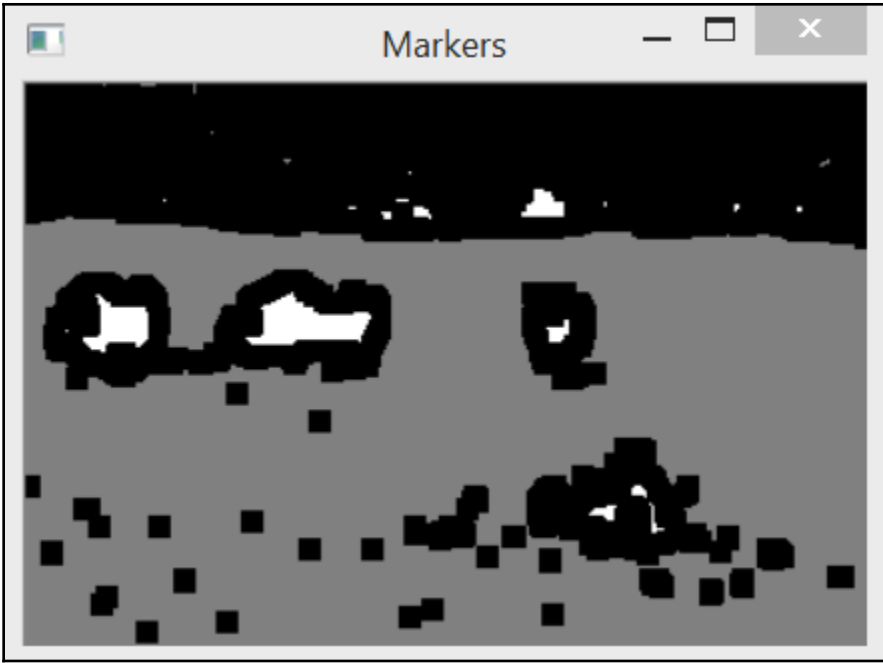


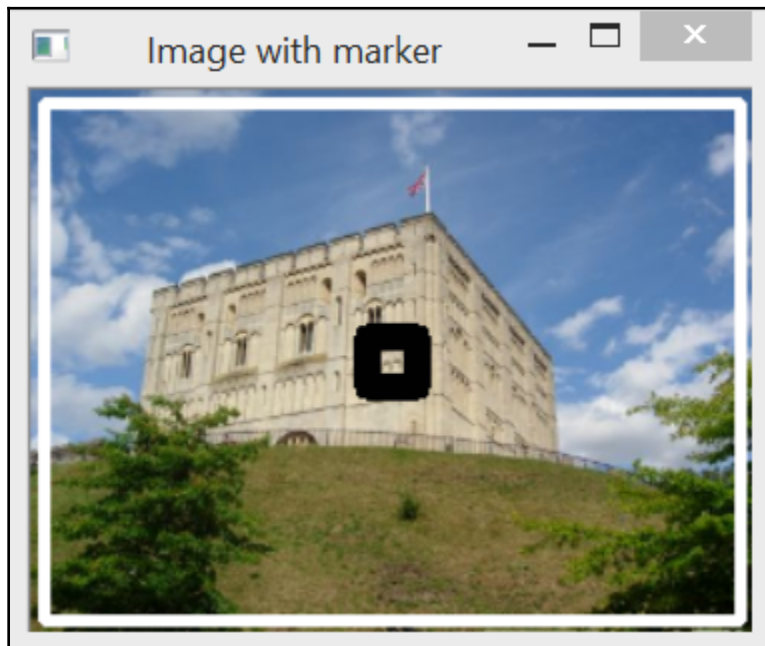
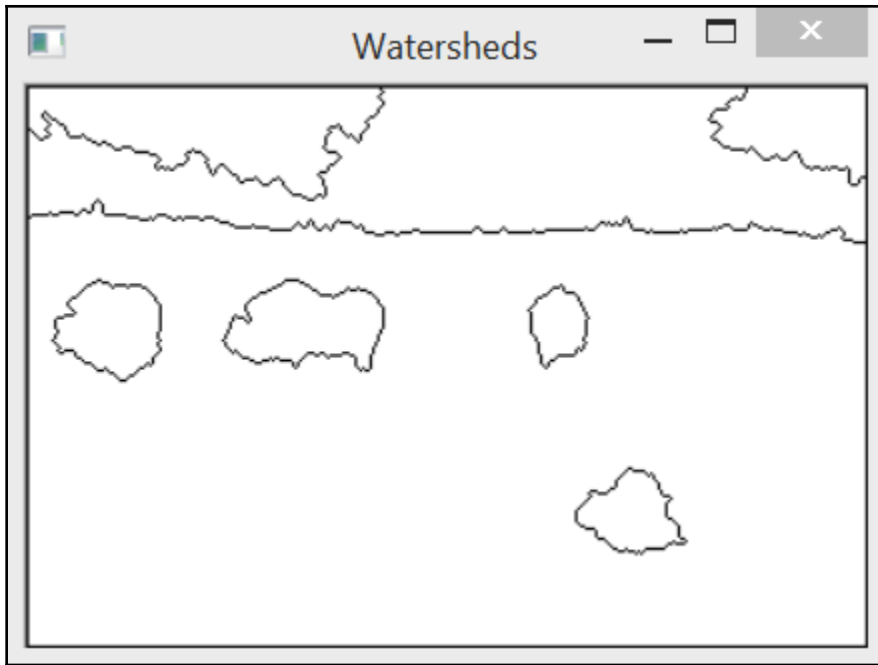


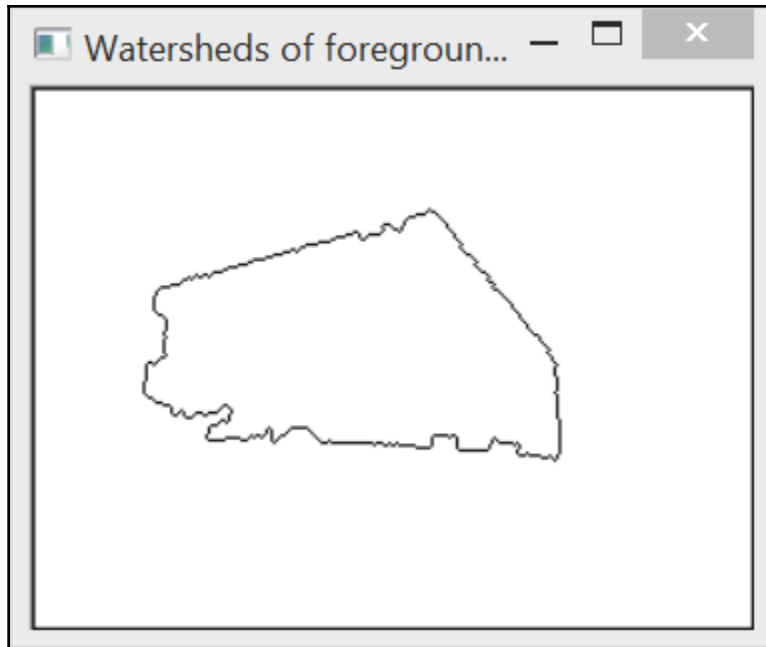


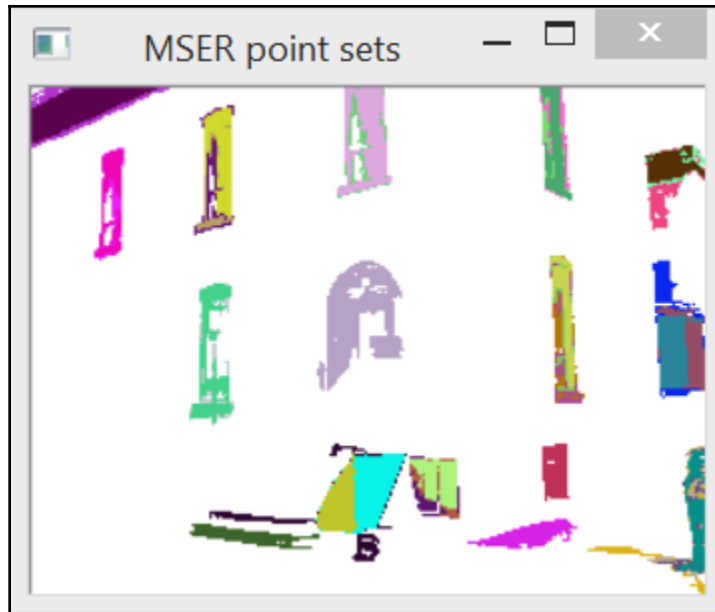


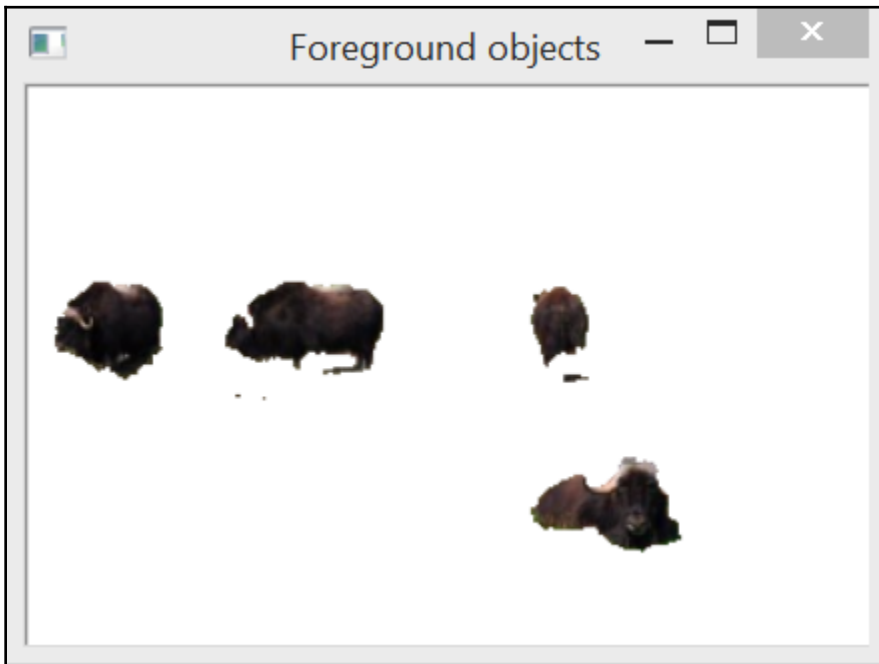
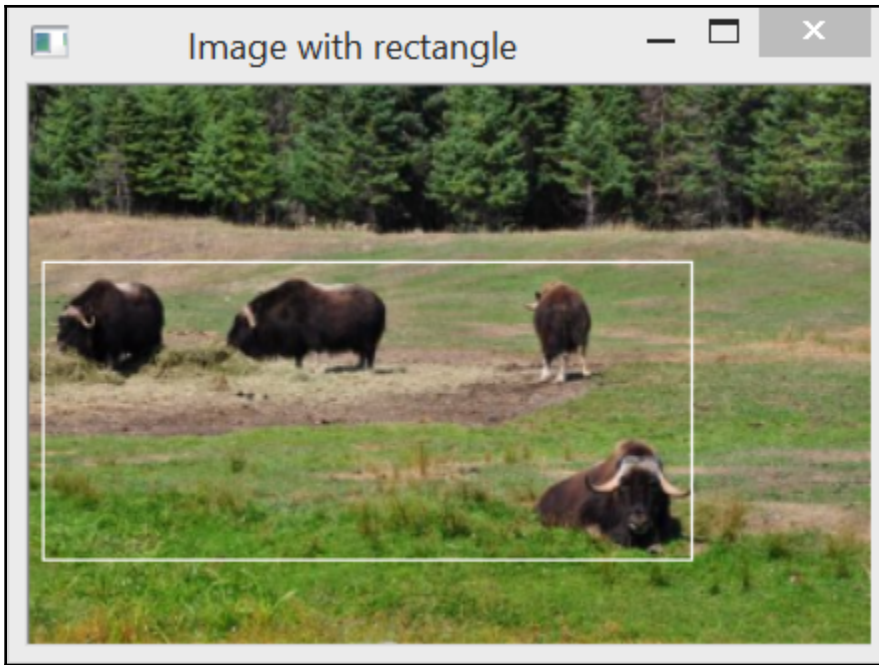




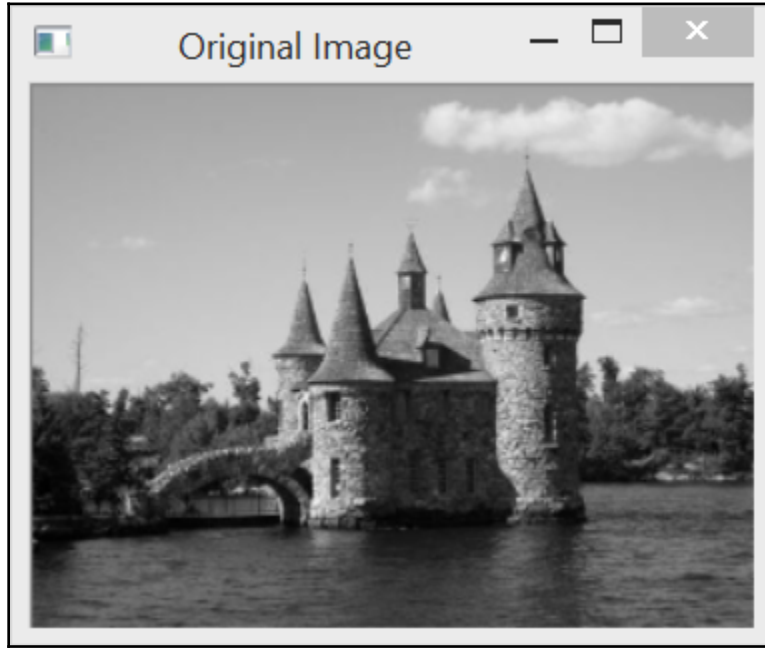


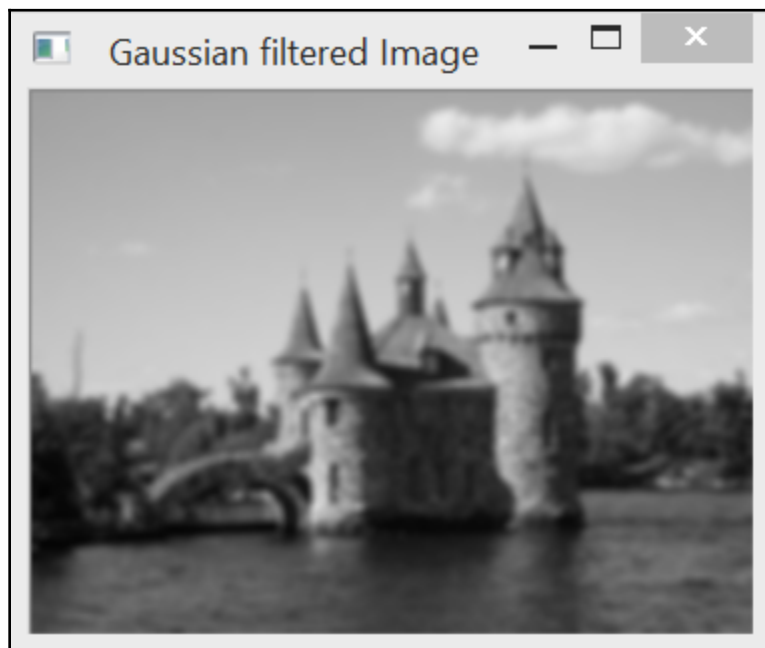
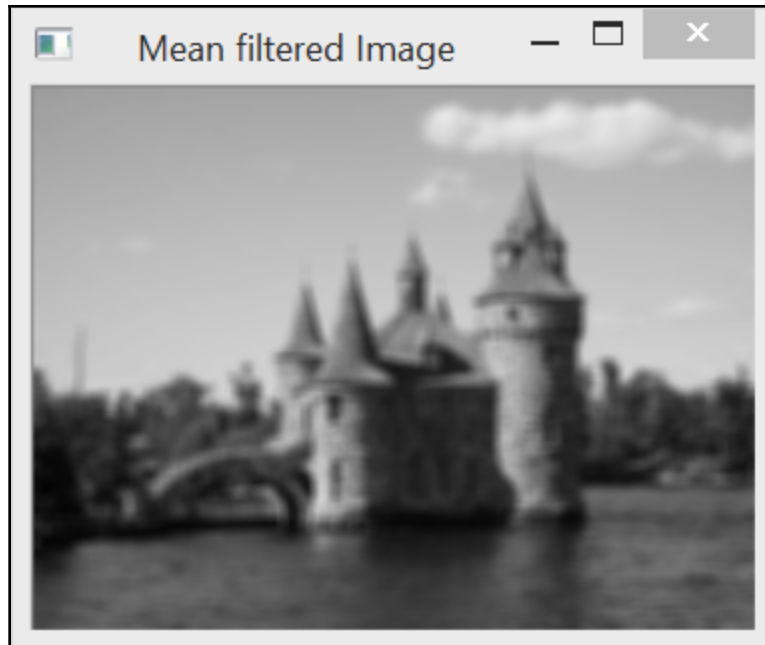


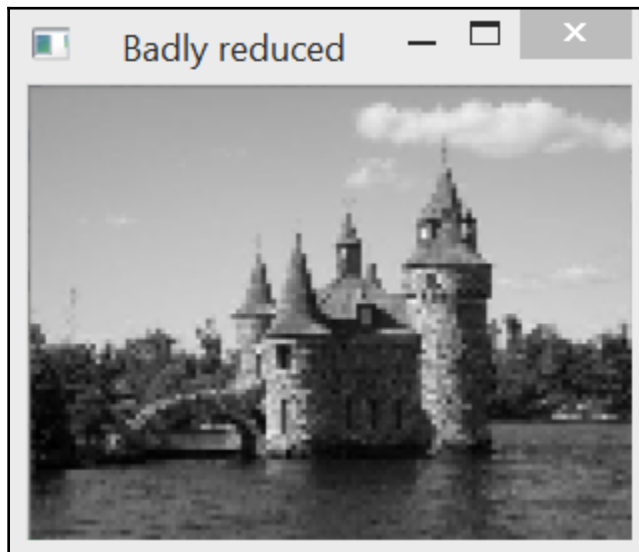
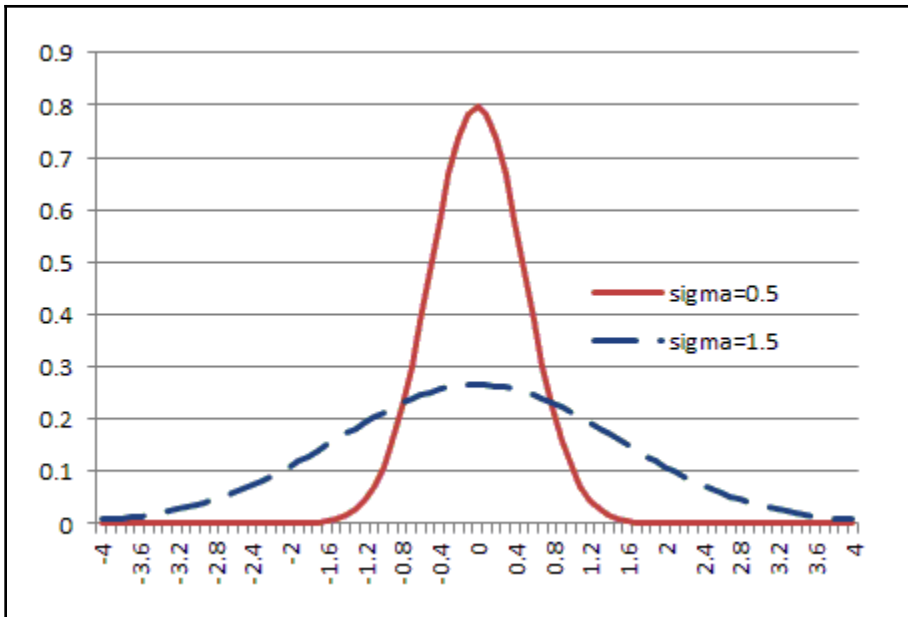


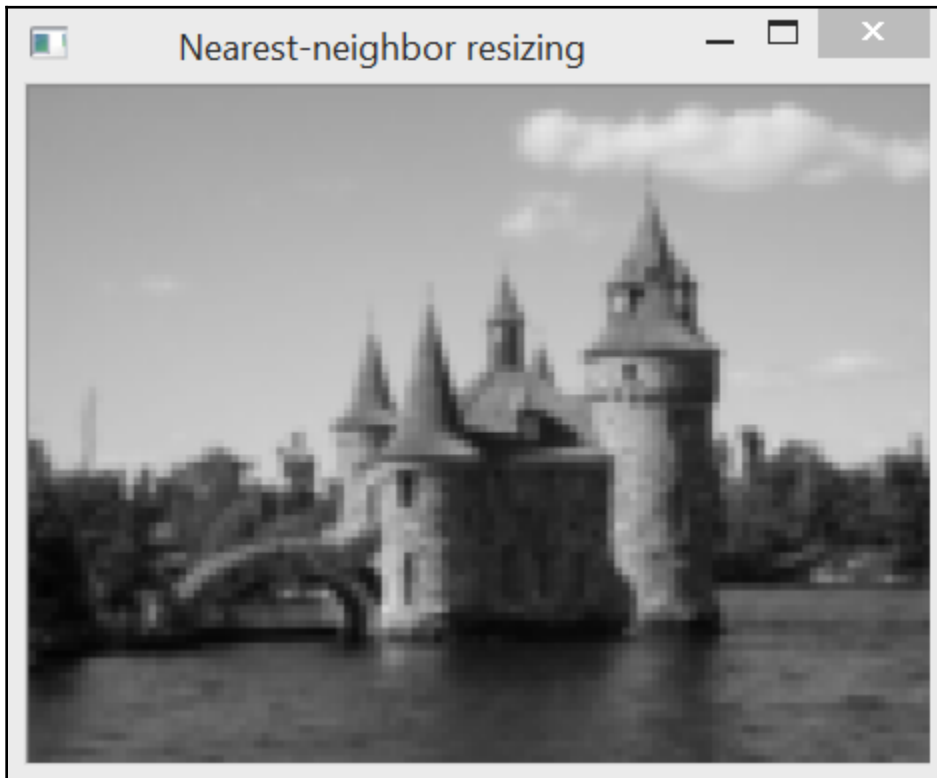
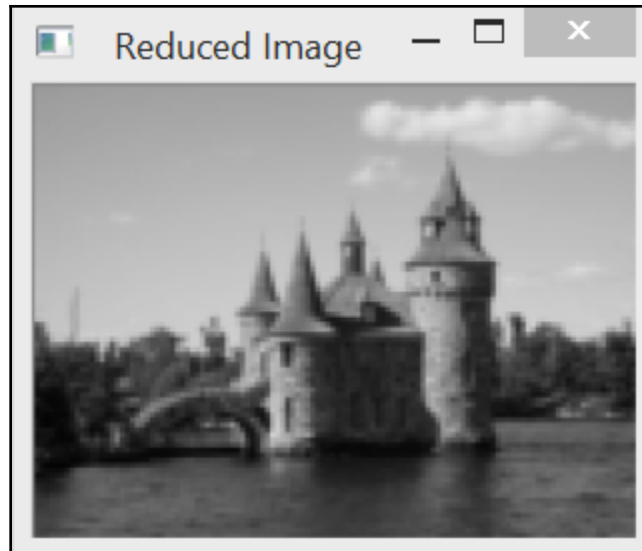


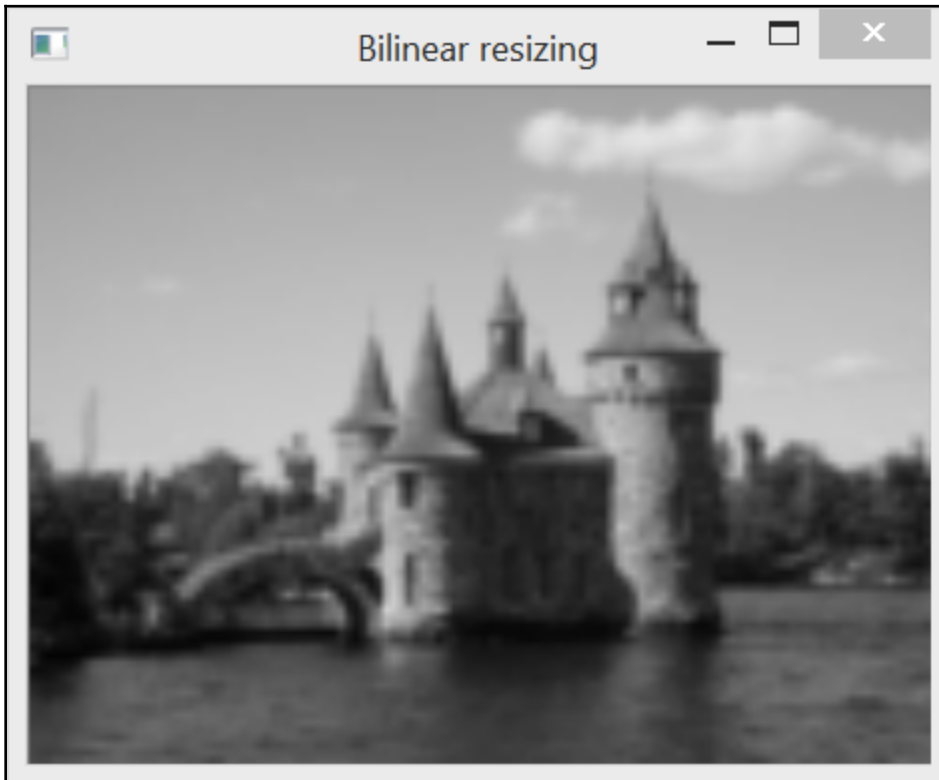
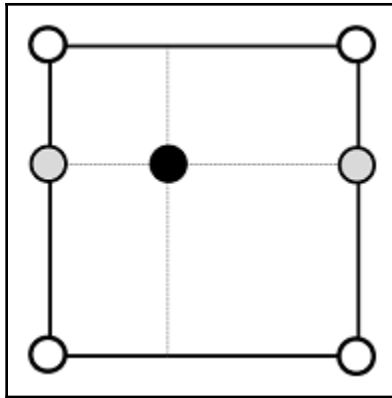
Chapter 6: Filtering the Images

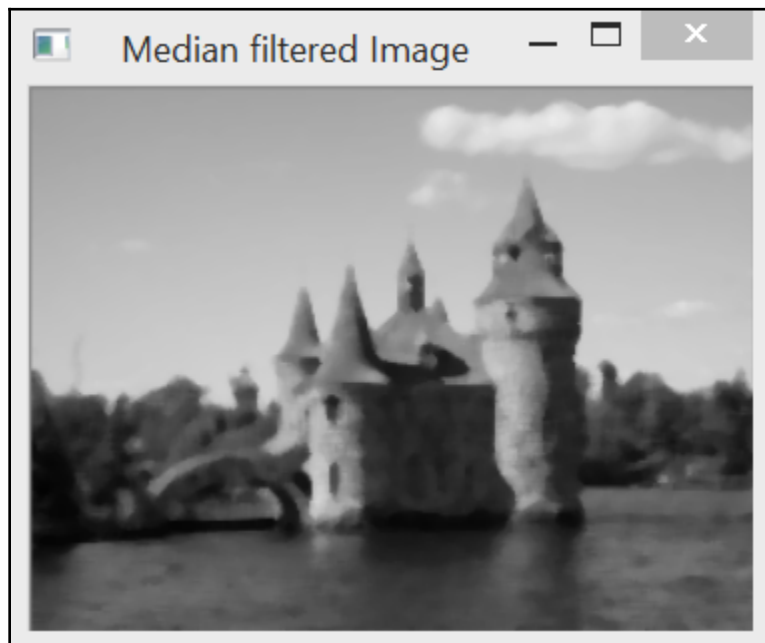
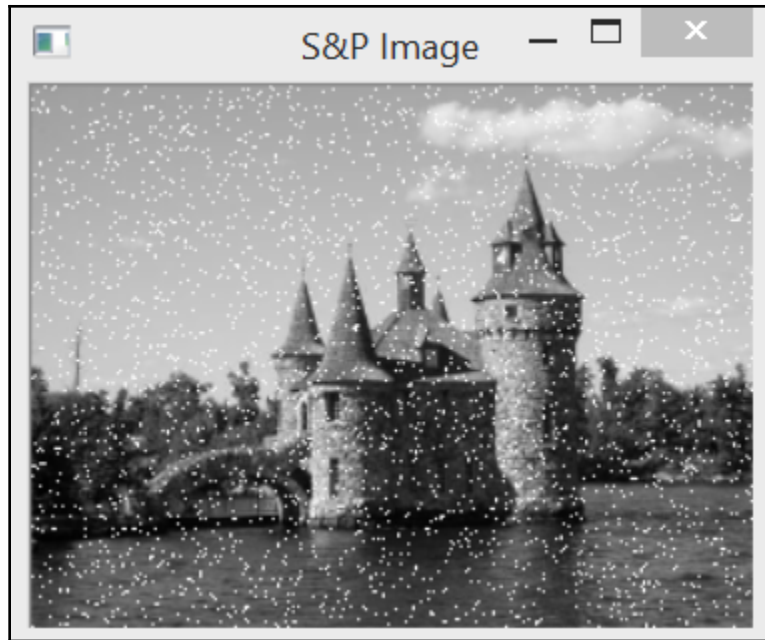


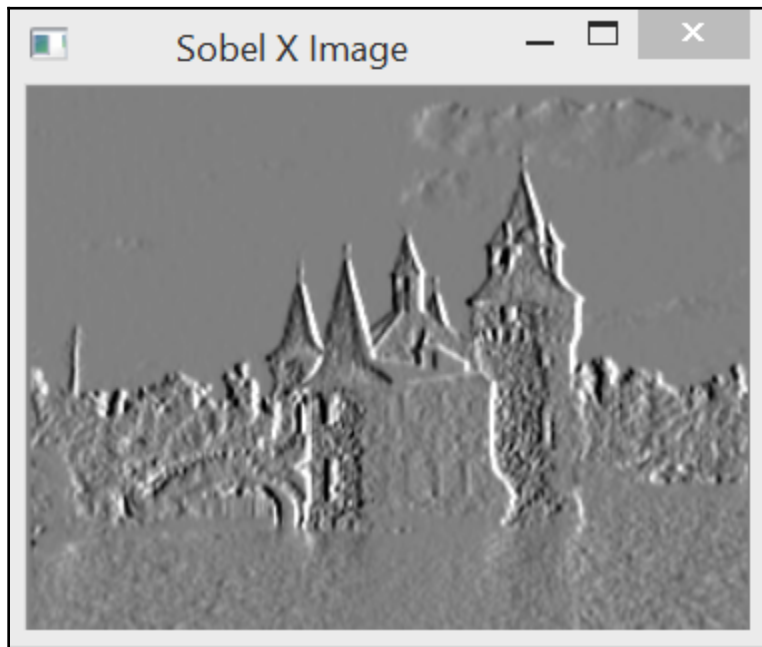
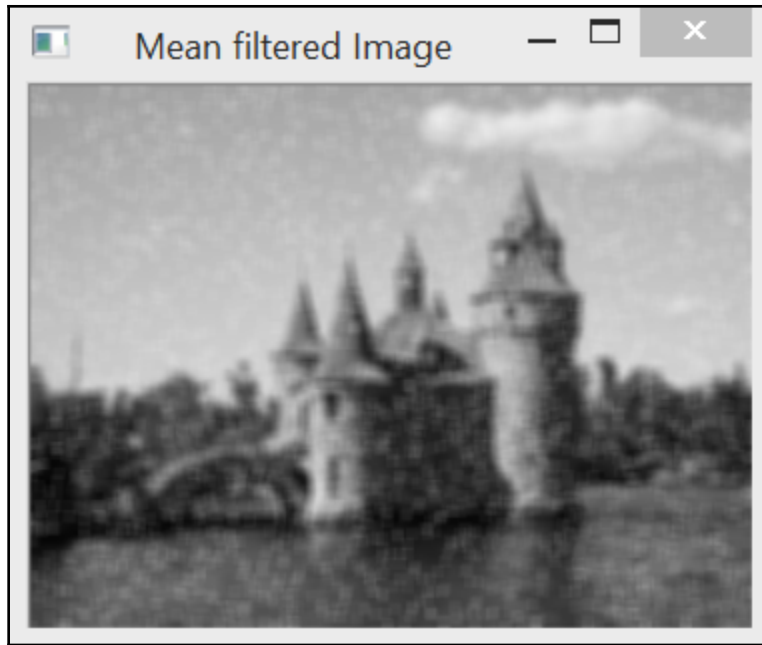


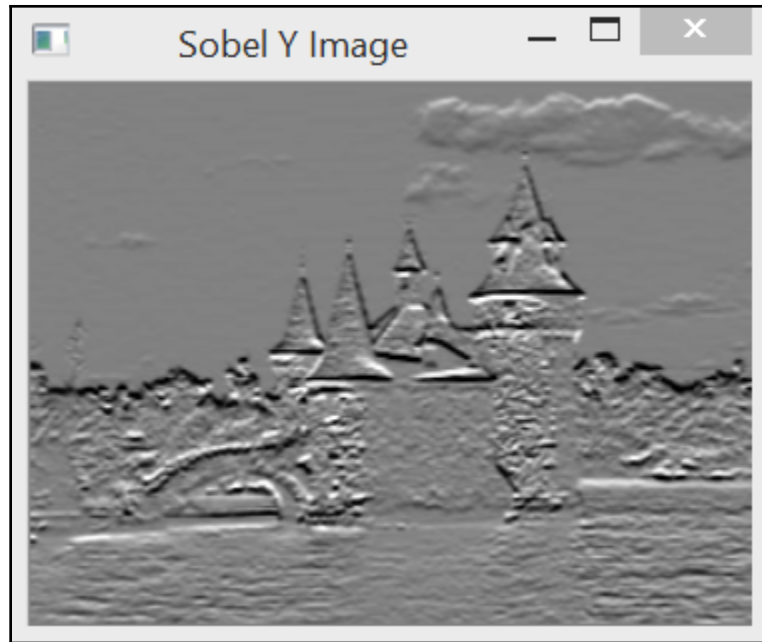


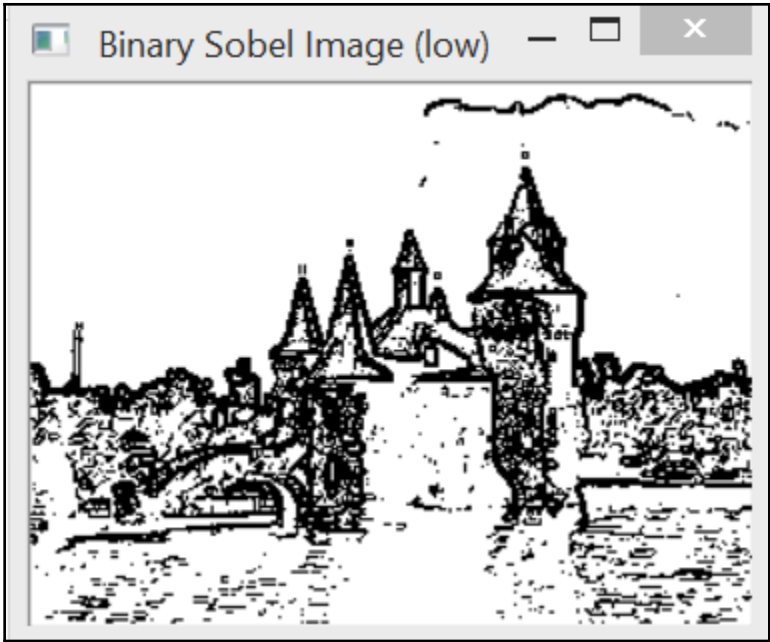


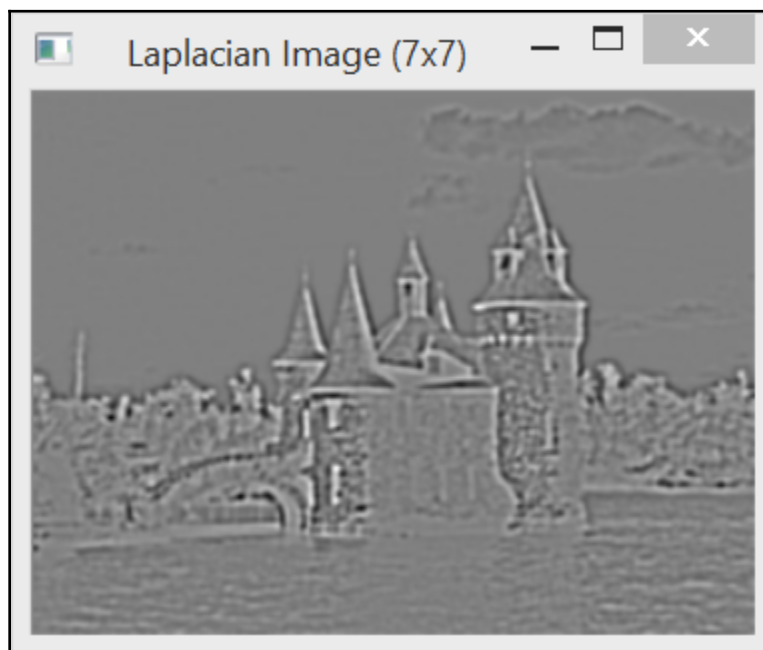
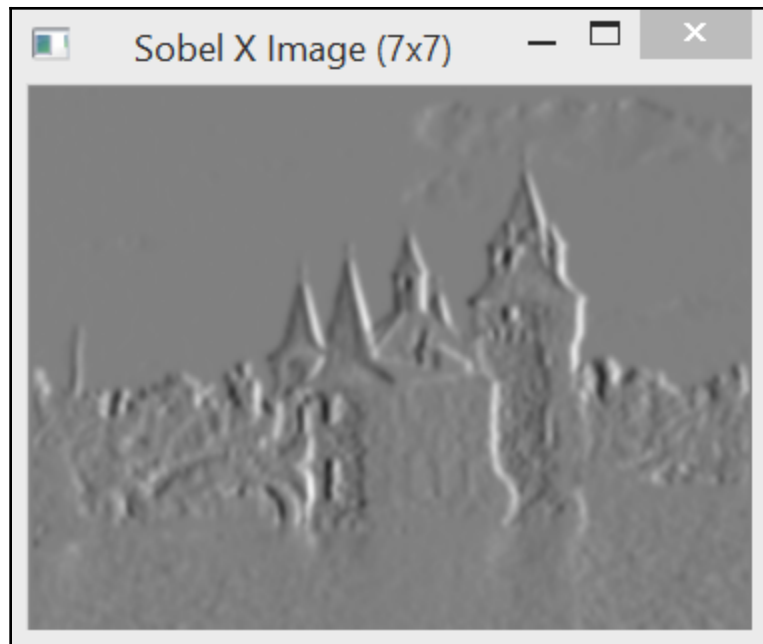


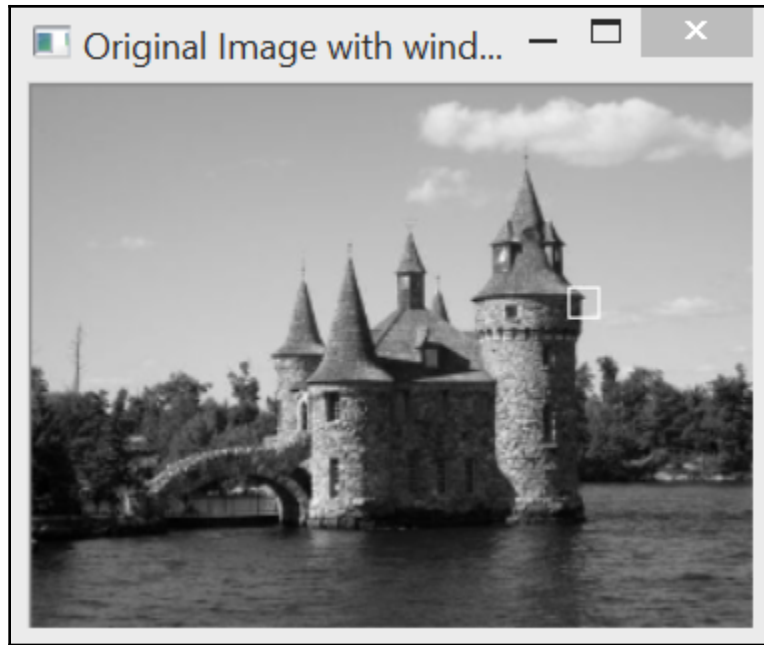










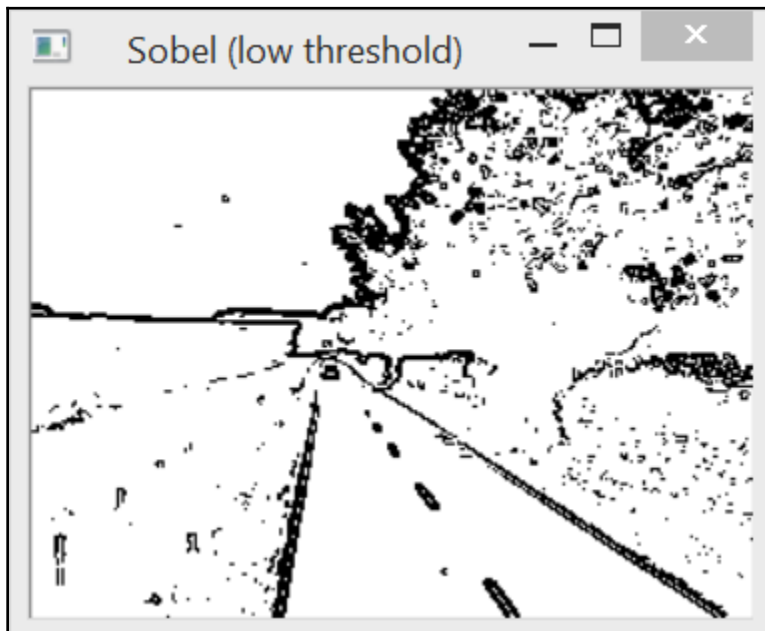
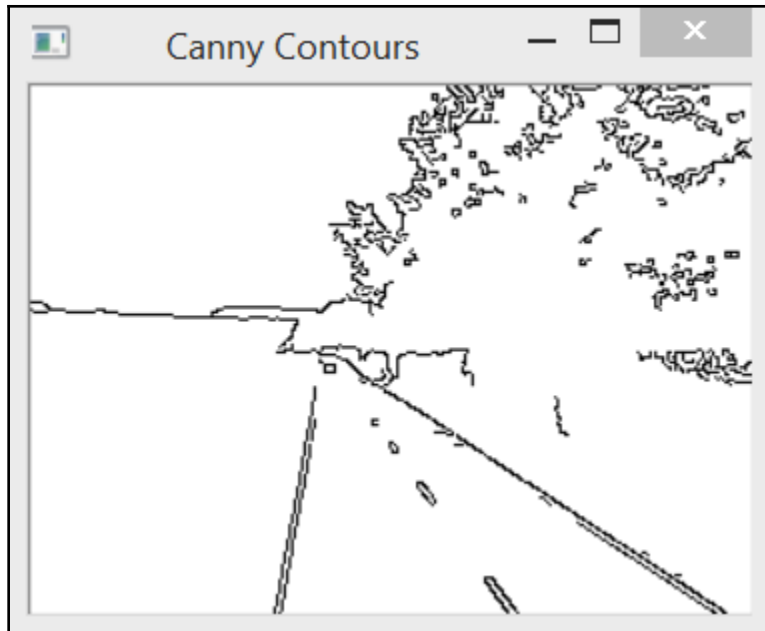


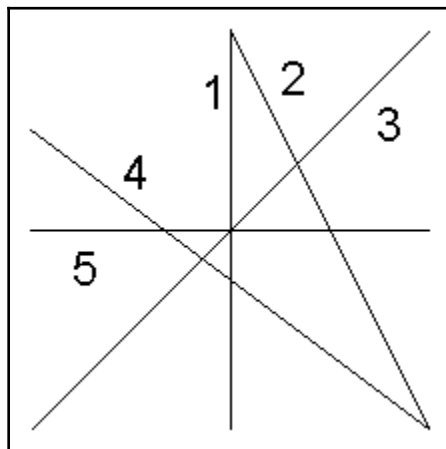
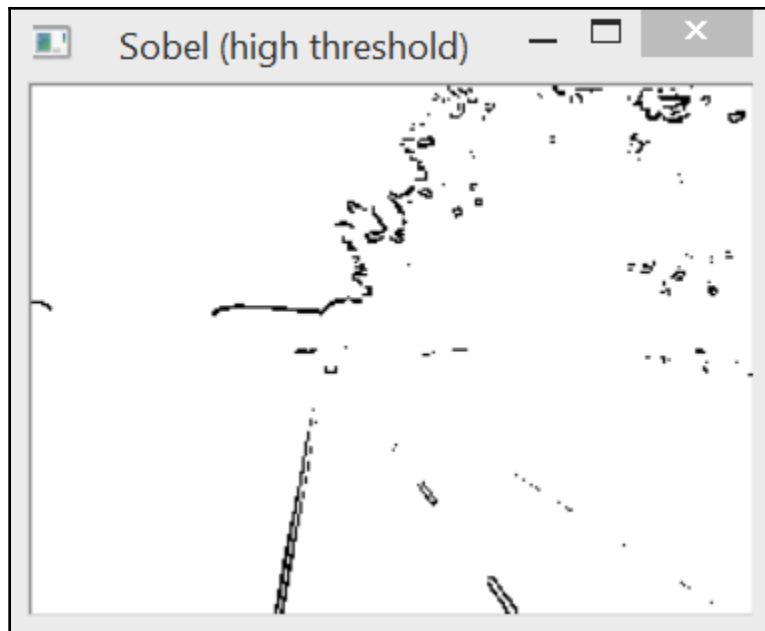
30	-128	-39	-12	-37	52	-9	0	0	0	0	-1
123	12	-97	-16	38	-43	8	2	0	0	0	0
-59	76	67	-102	-32	15	-5	-8	2	3	0	1
-82	-75	-48	72	-128	-28	-10	8	3	-10	4	-2
72	63	43	81	74	-128	-33	-12	1	-2	7	0
16	23	-12	31	127	127	-58	-11	11	-1	-11	2
-3	-22	71	48	-11	-128	-12	-10	0	0	7	-4
3	10	-17	75	-14	-86	-2	5	1	-4	0	0
-1	2	-14	46	-10	-44	1	2	2	4	2	5
-26	-25	16	112	-30	-96	2	-3	-2	-6	-6	-6
-12	-11	6	97	-28	-62	4	1	4	8	9	8
8	-4	9	117	-8	-71	6	3	6	1	3	3



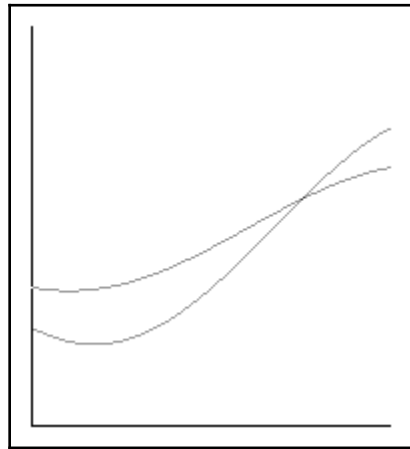
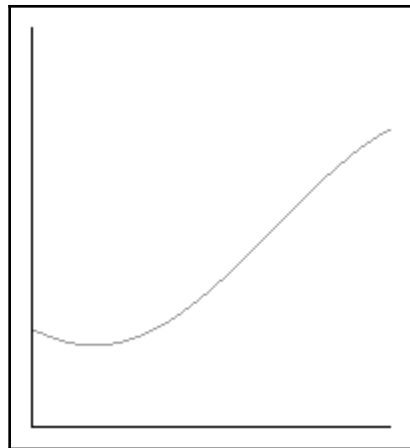
Chapter 7: Extracting Lines, Contours, and Components



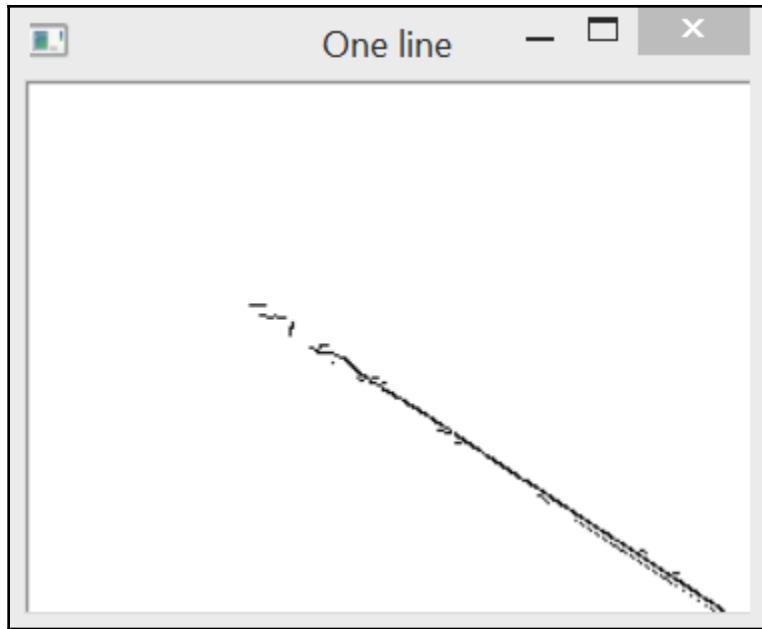


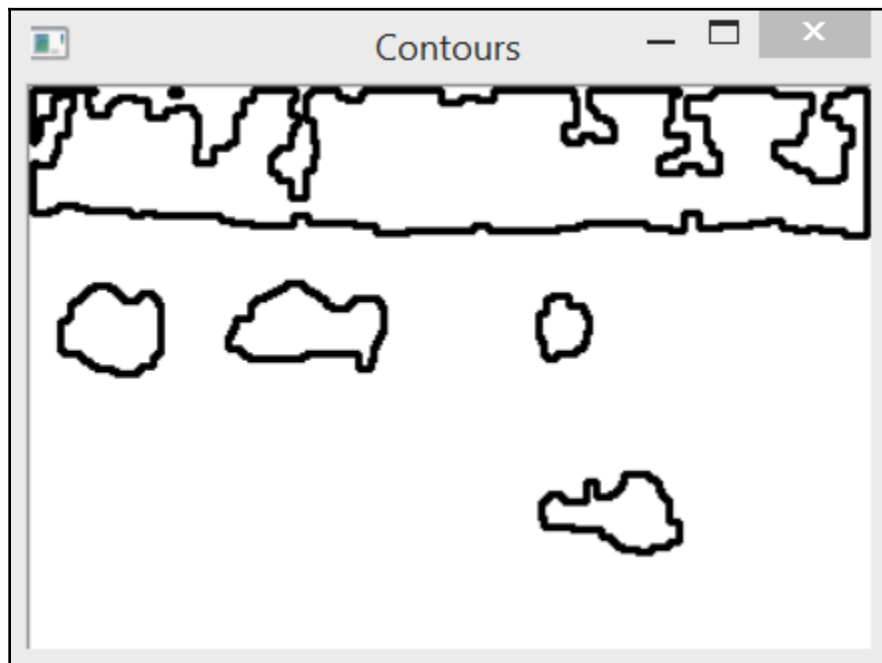
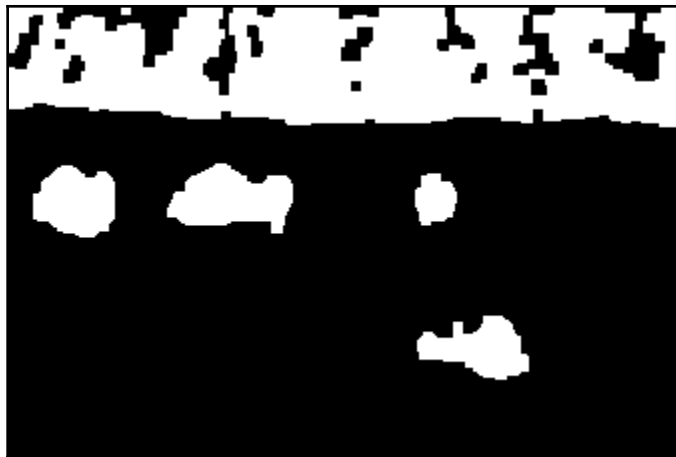


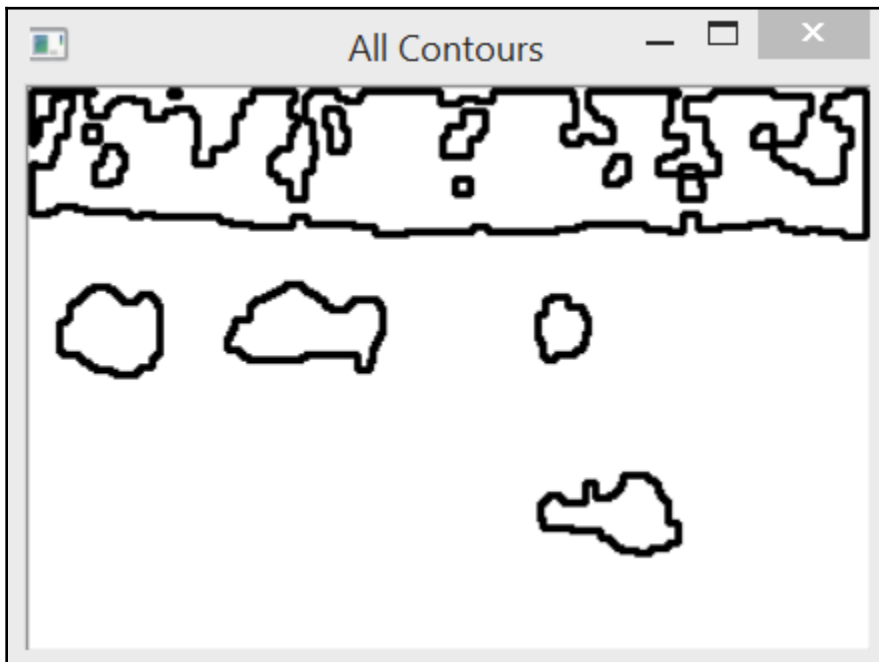
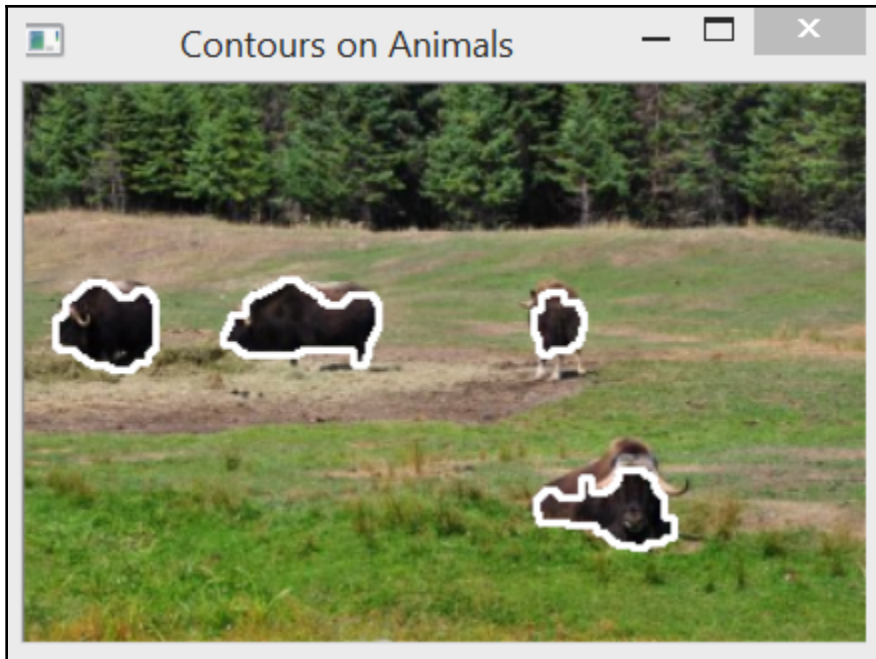


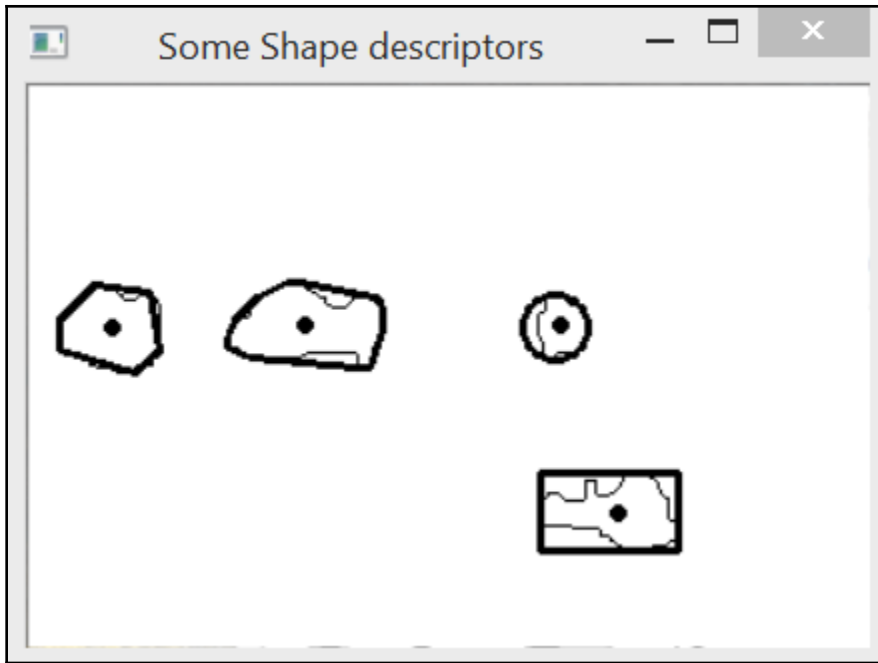


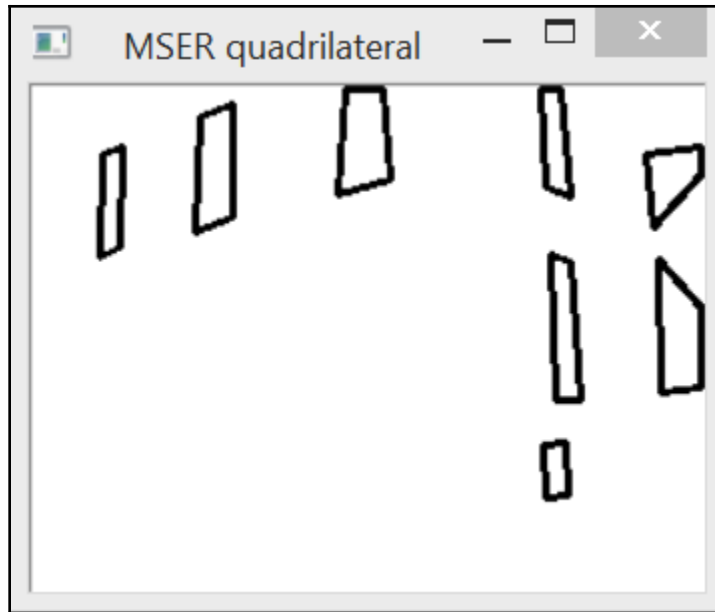






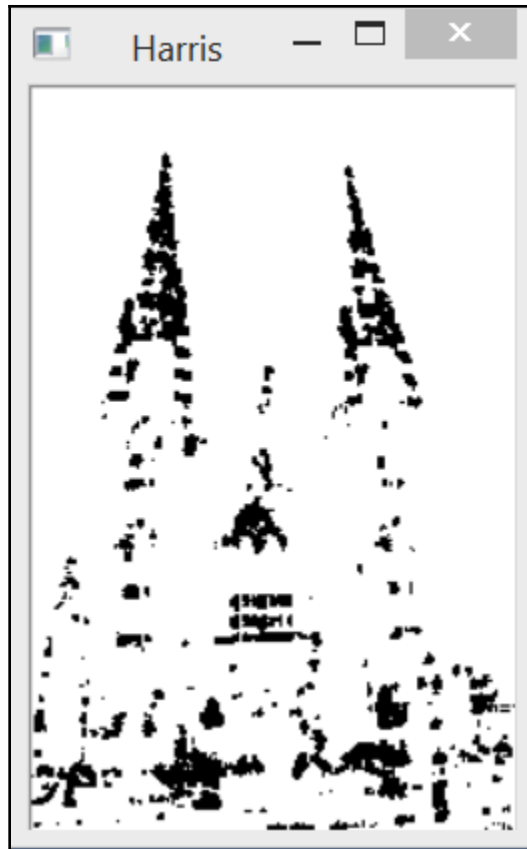






Chapter 8: Detecting Interest Points



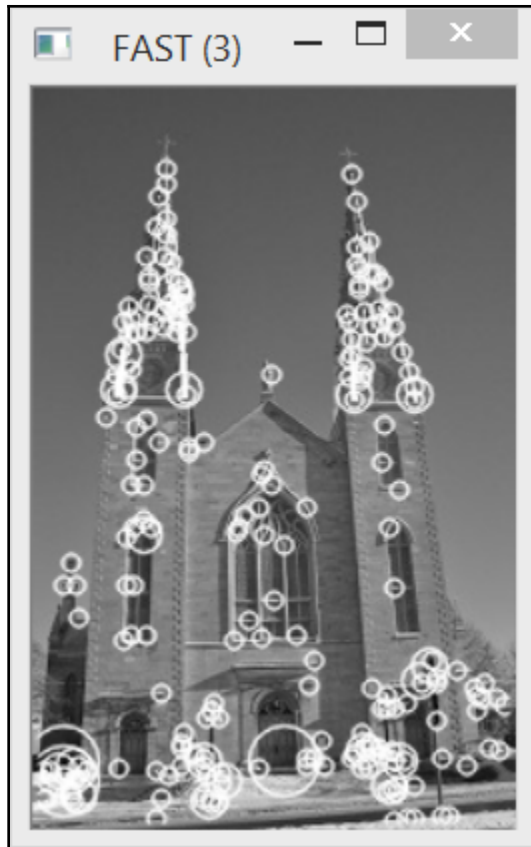




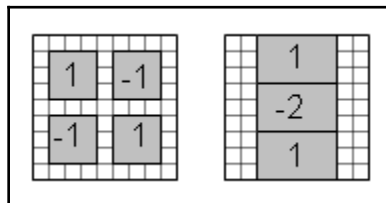


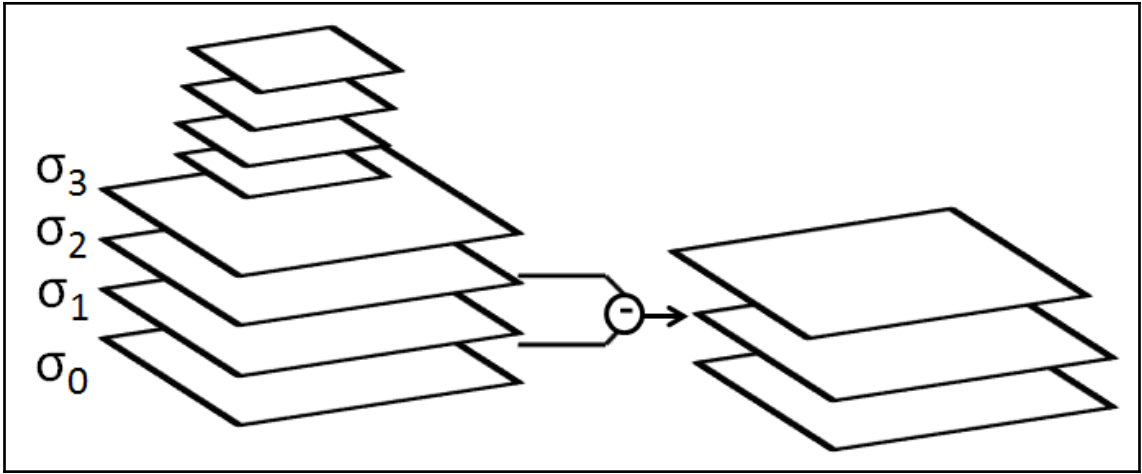


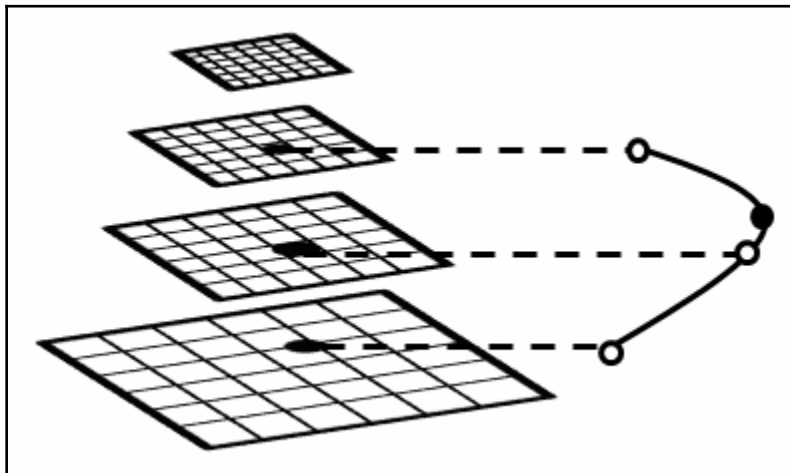
		16	1	2		
	15				3	
14						4
13			0			5
12						6
	11				7	
		10	9	8		





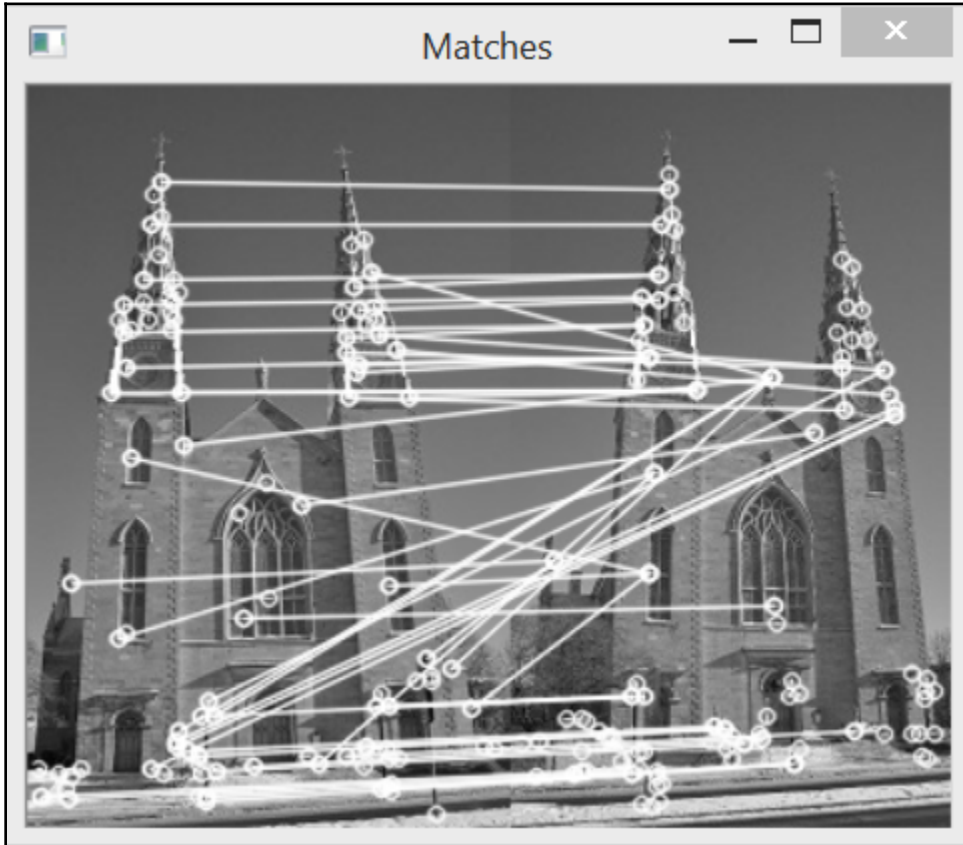


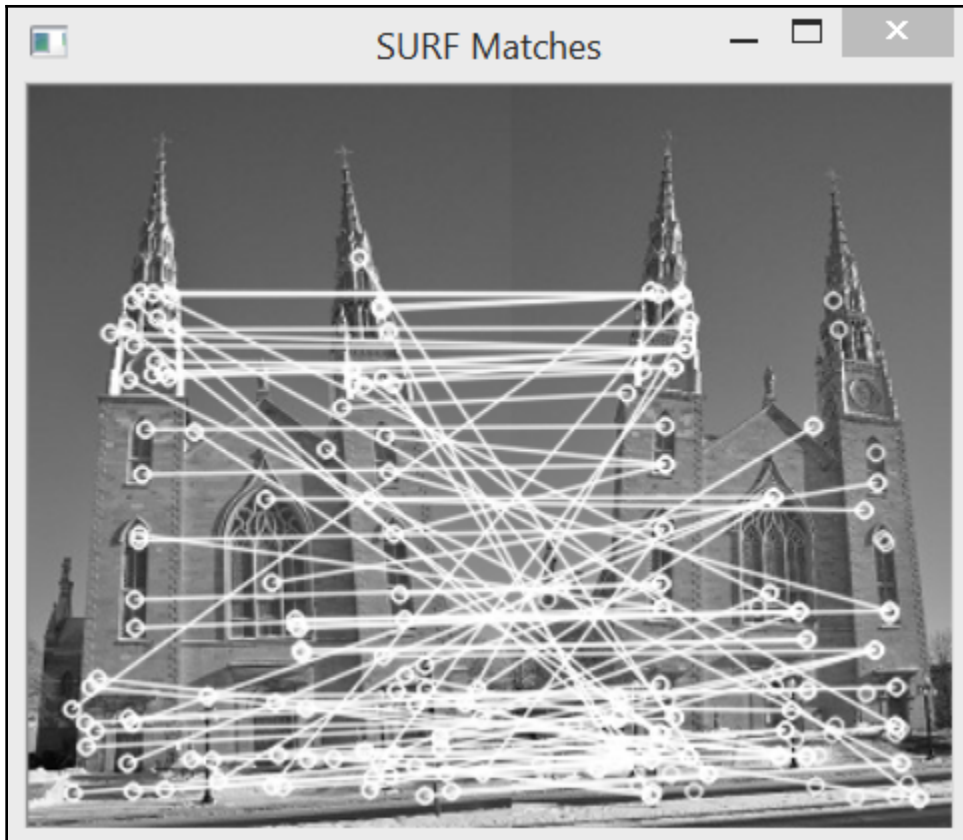


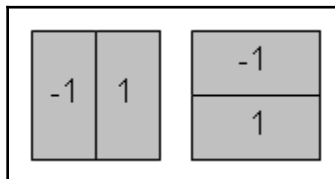


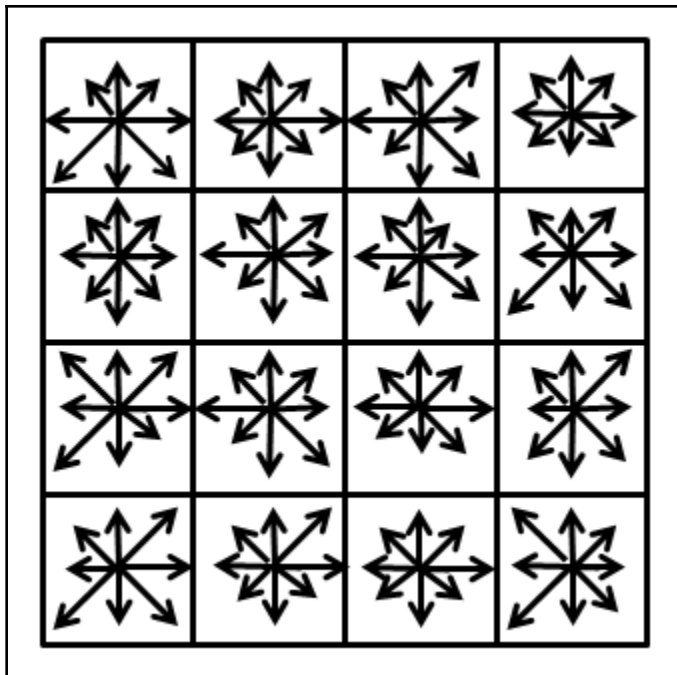


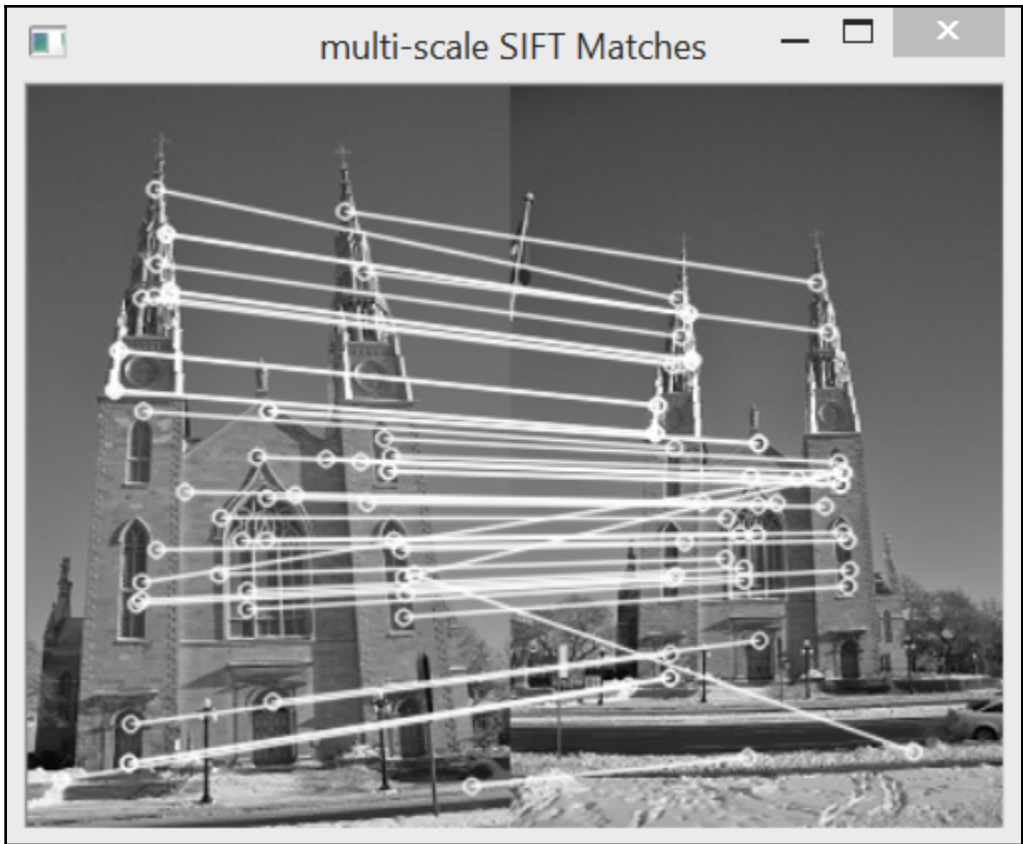
Chapter 9: Describing and Matching Interest Points

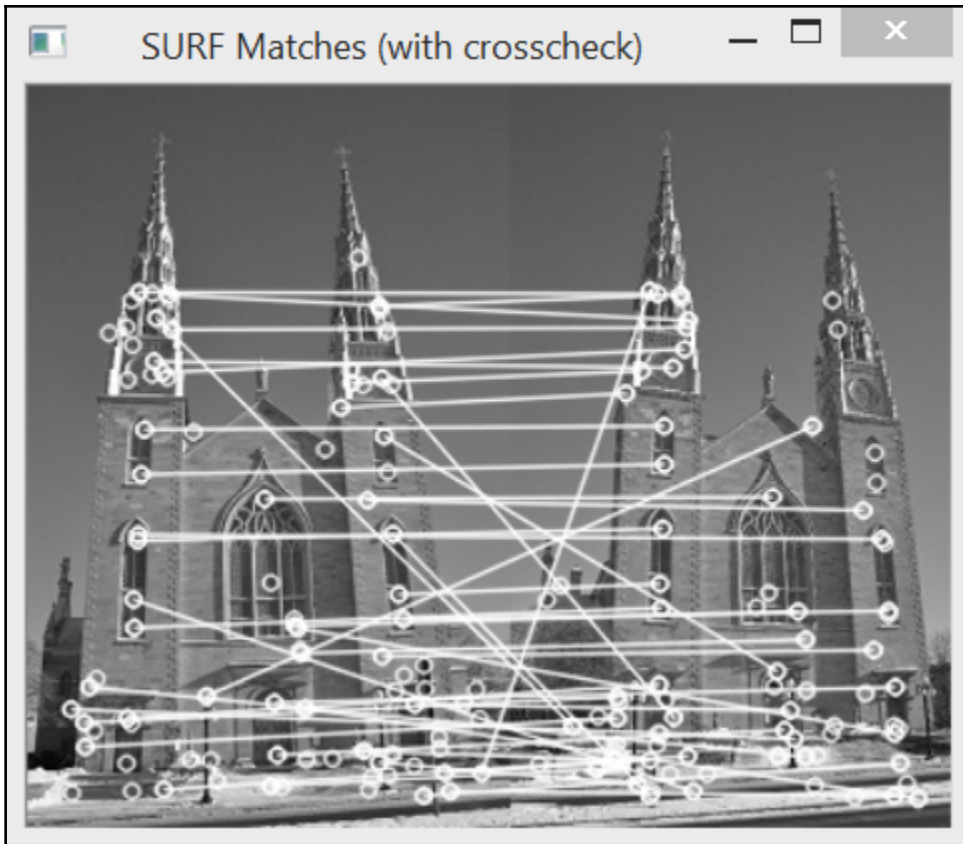


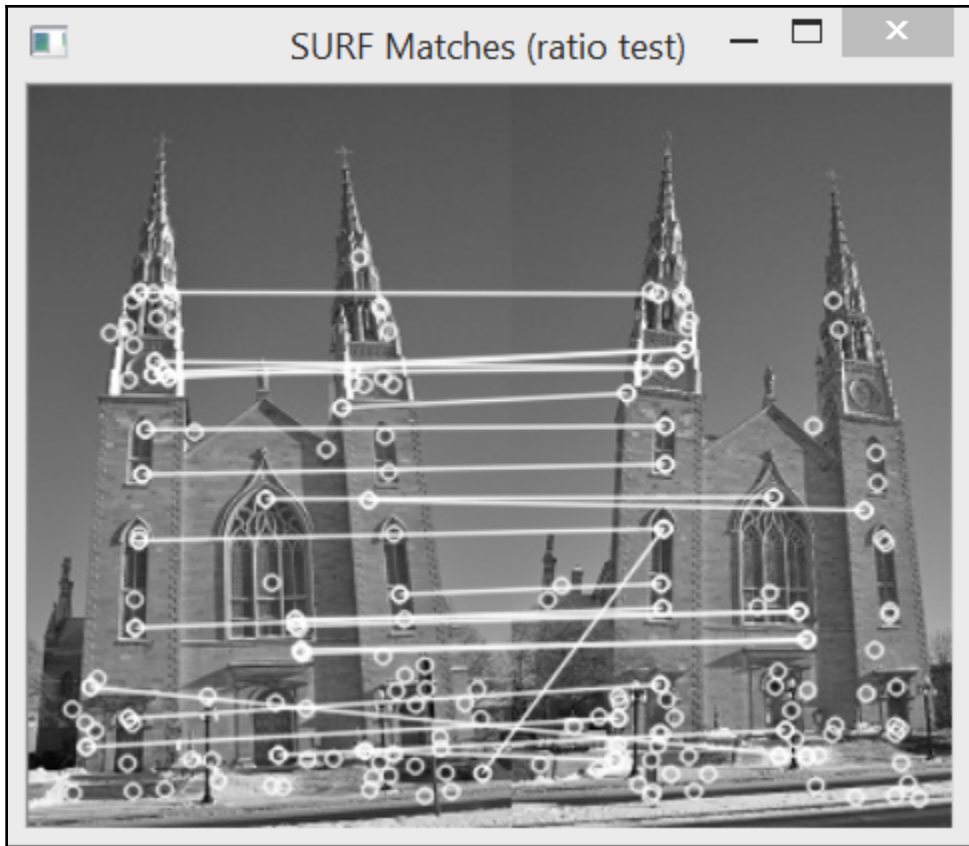


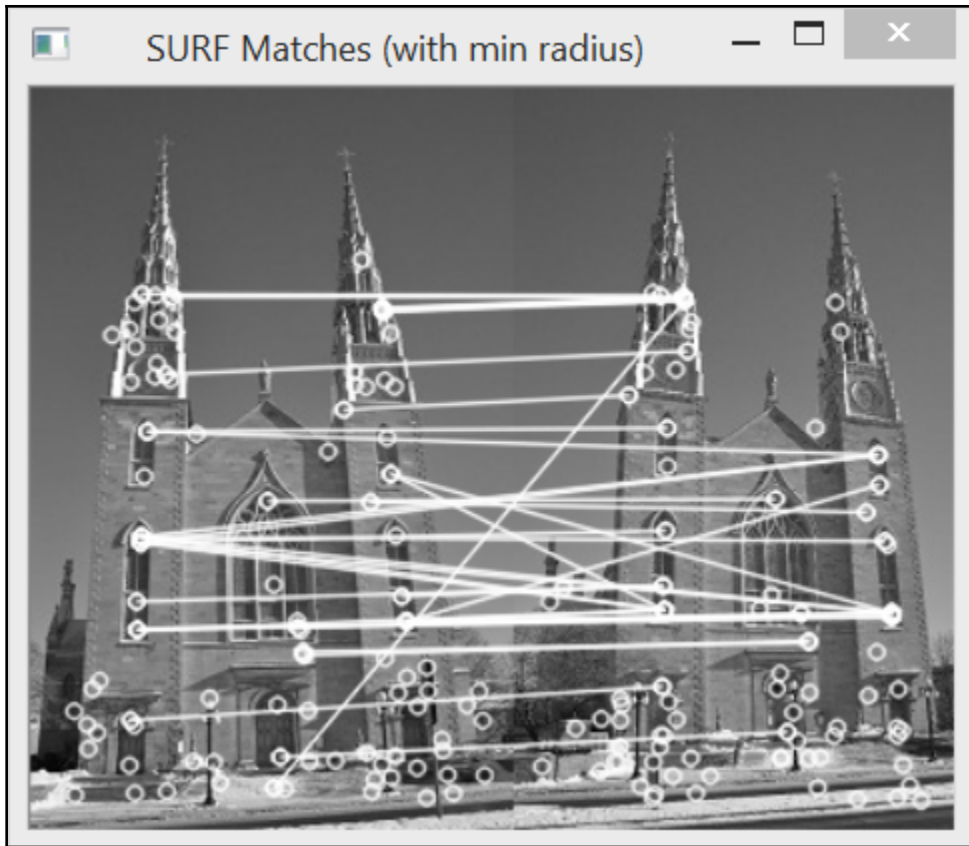


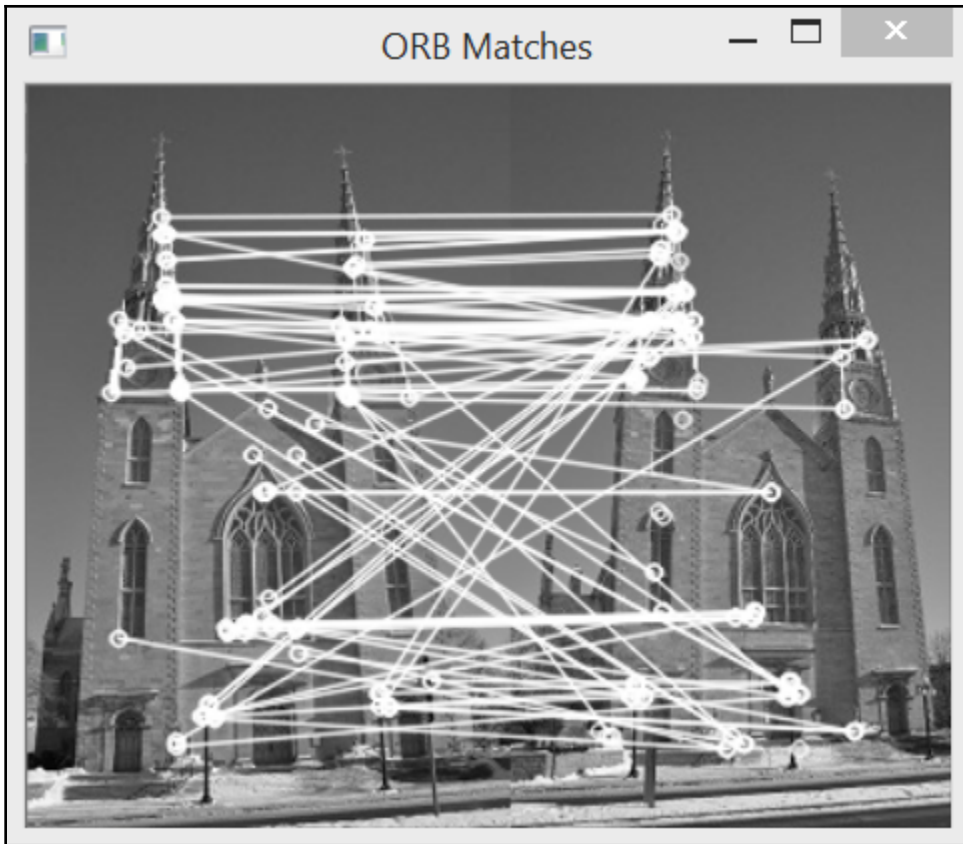


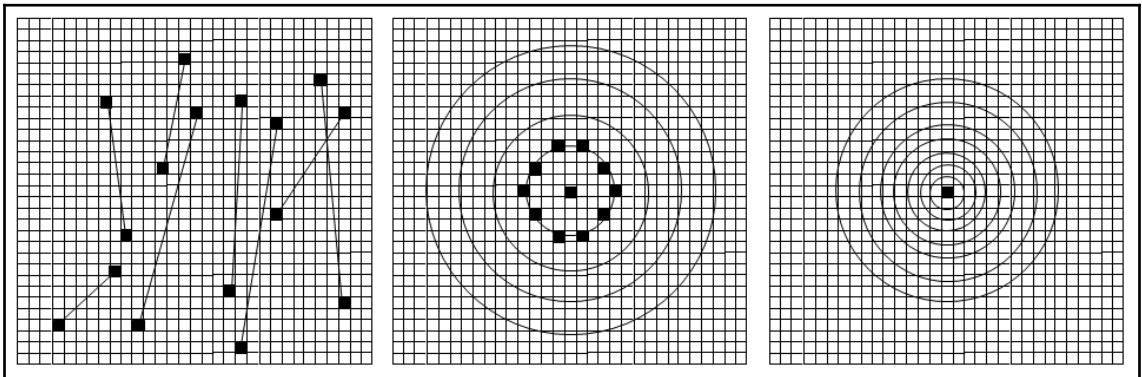




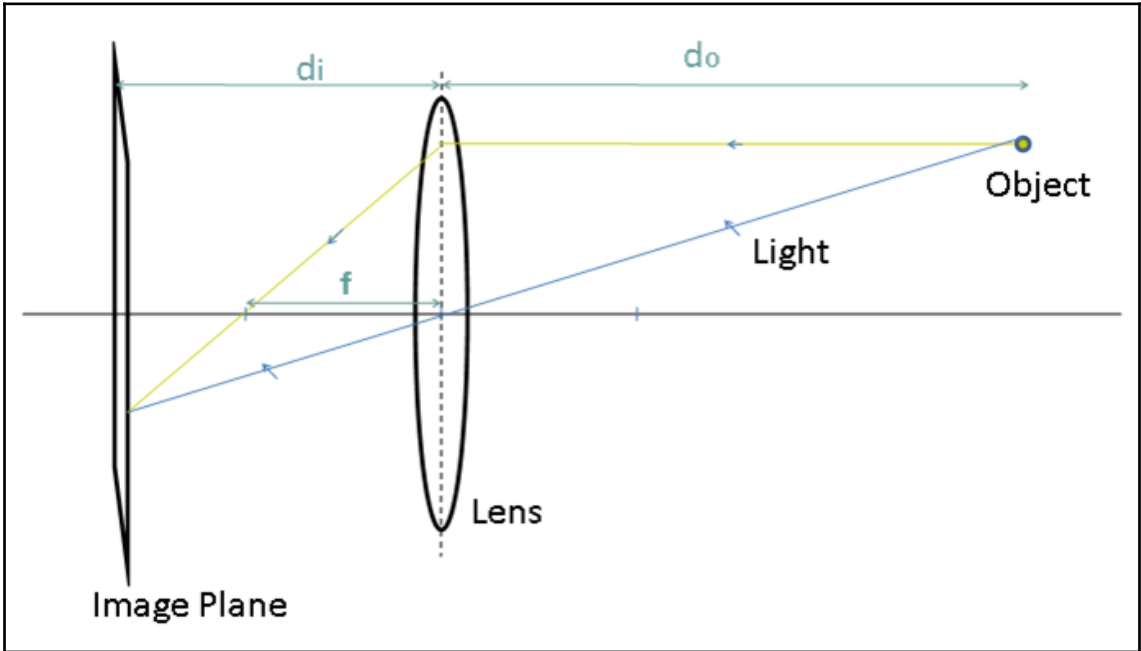


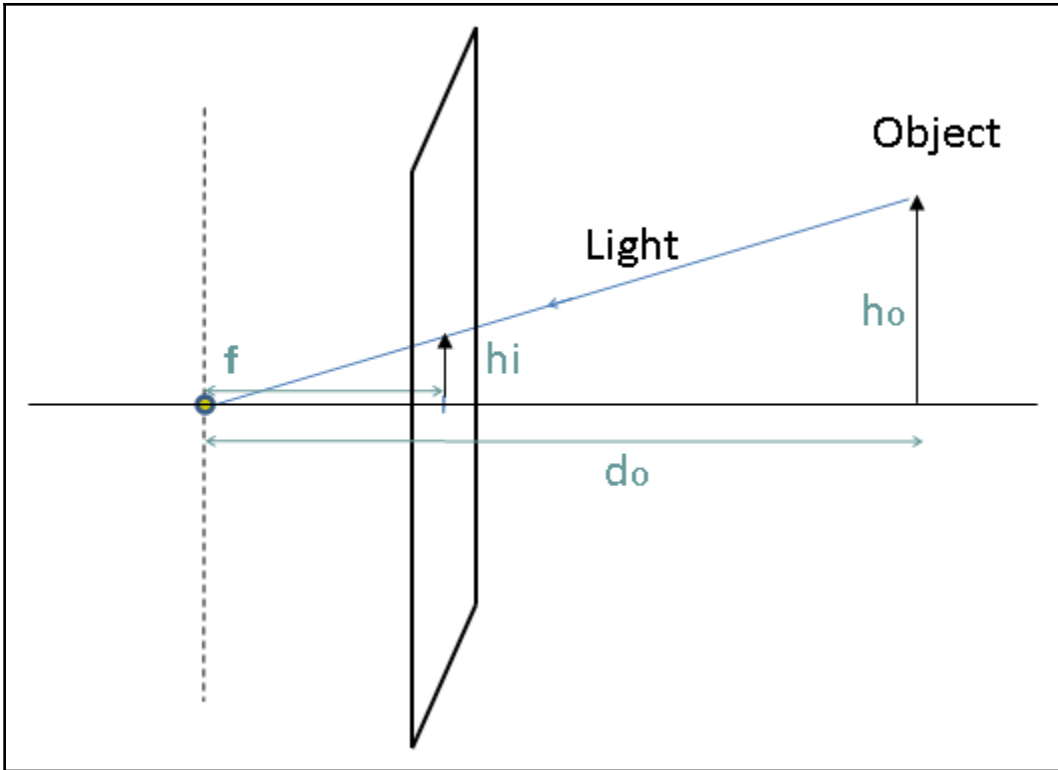


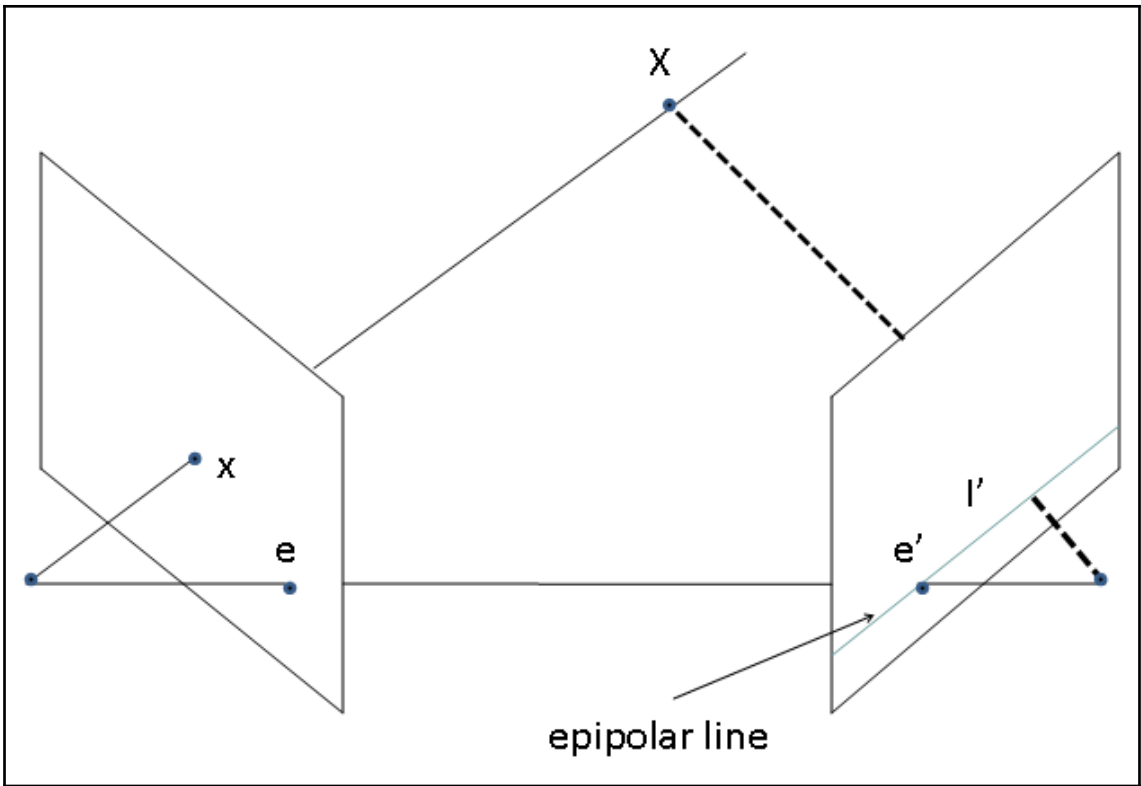


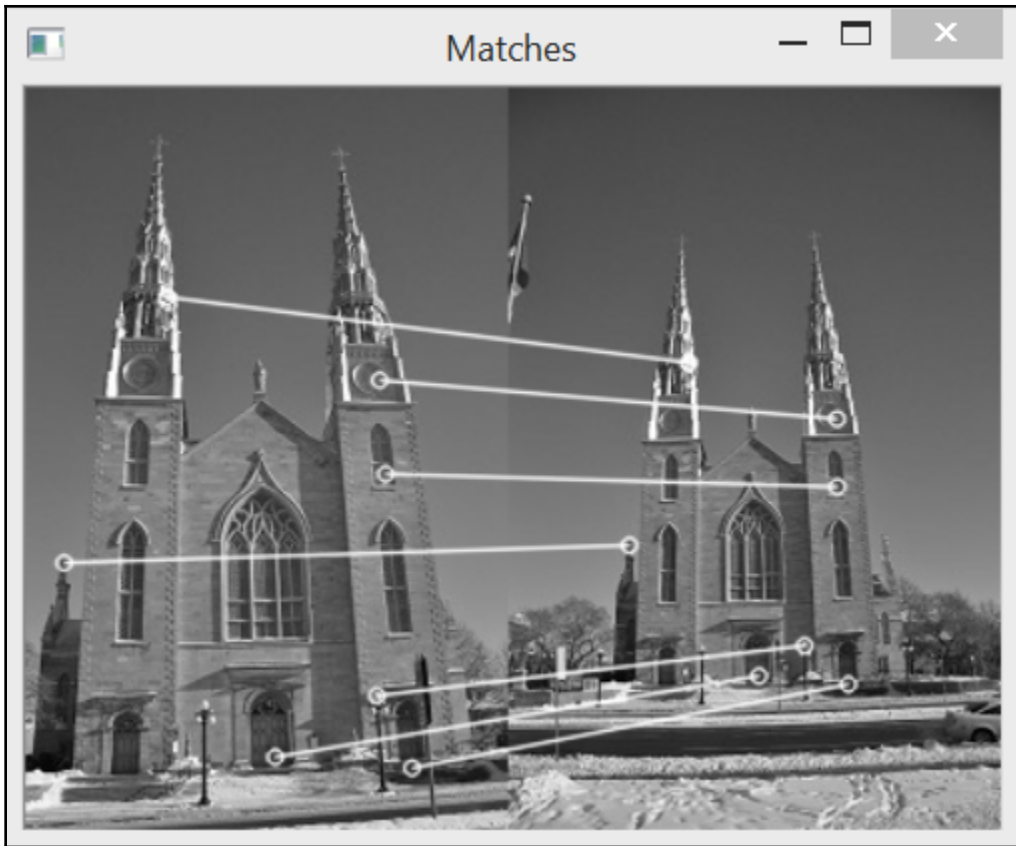


Chapter 10: Estimating Projective Relations in Images



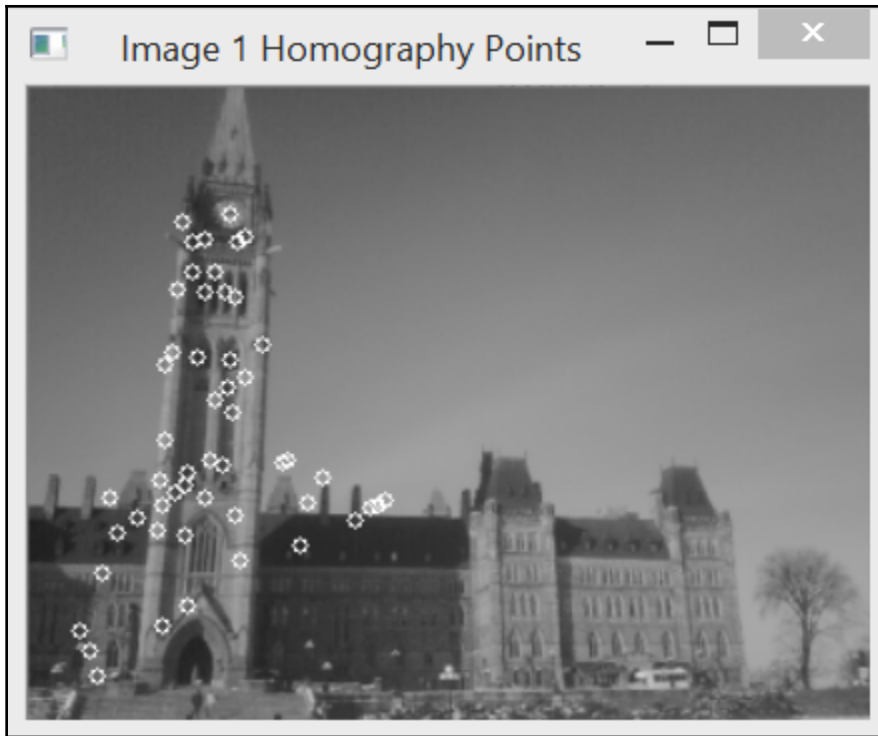


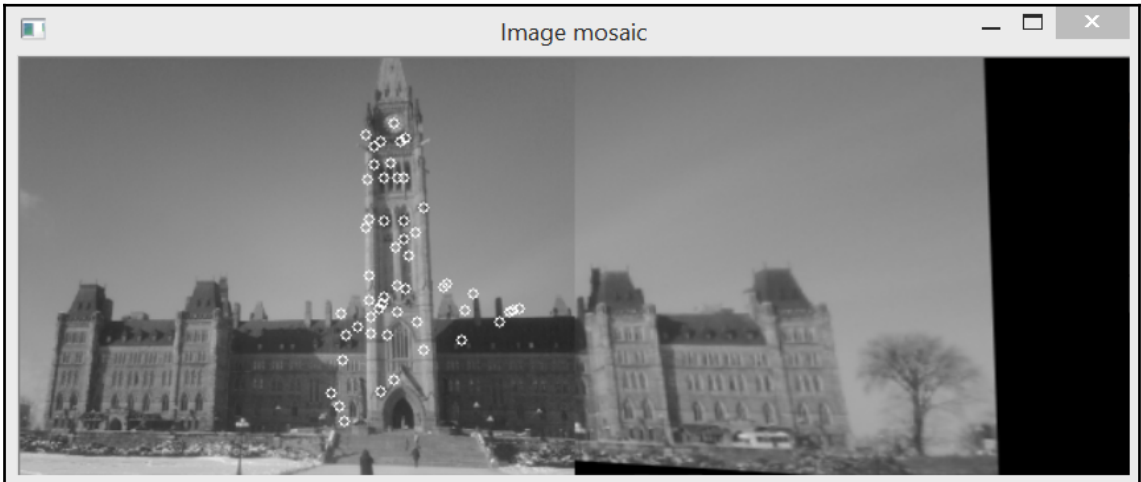


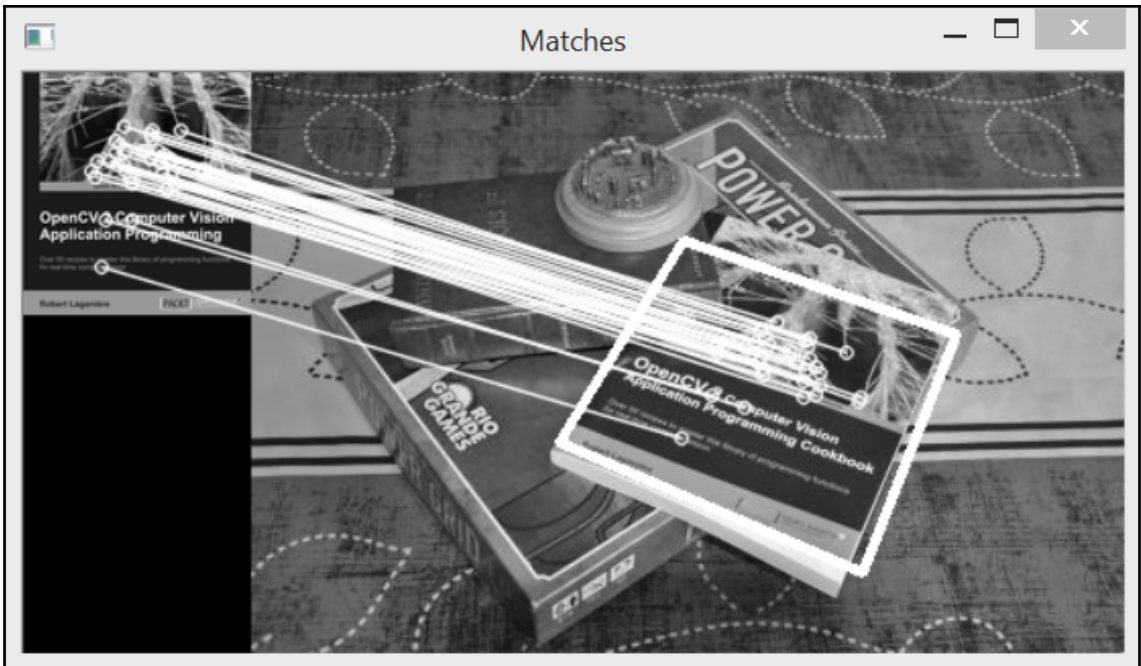




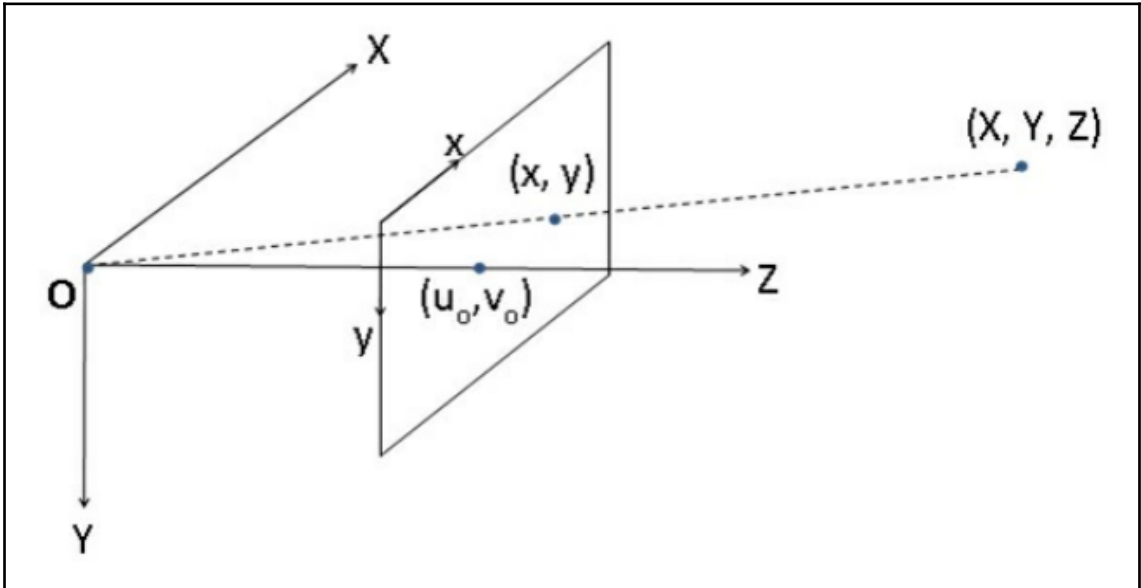


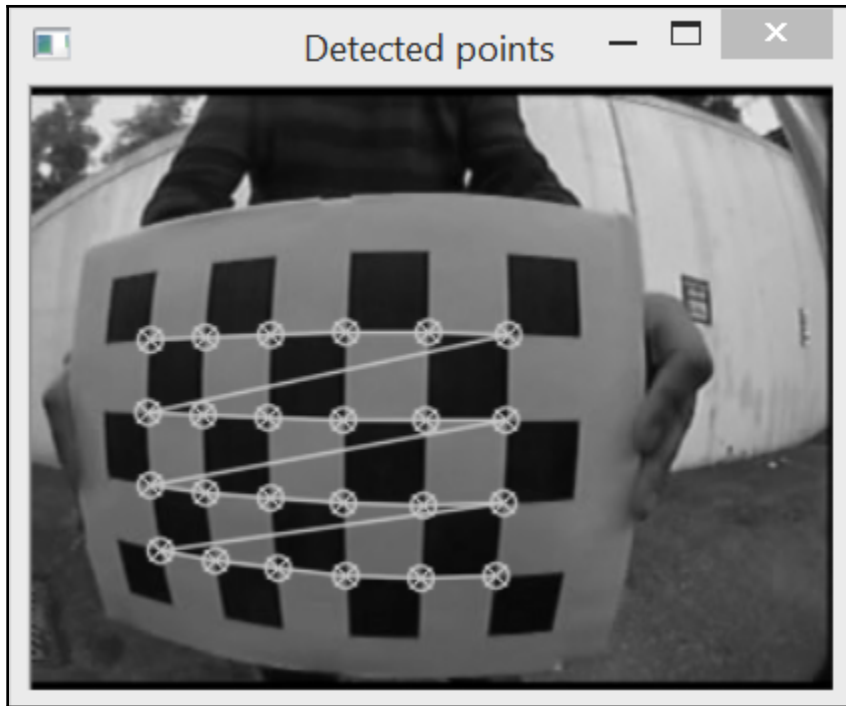


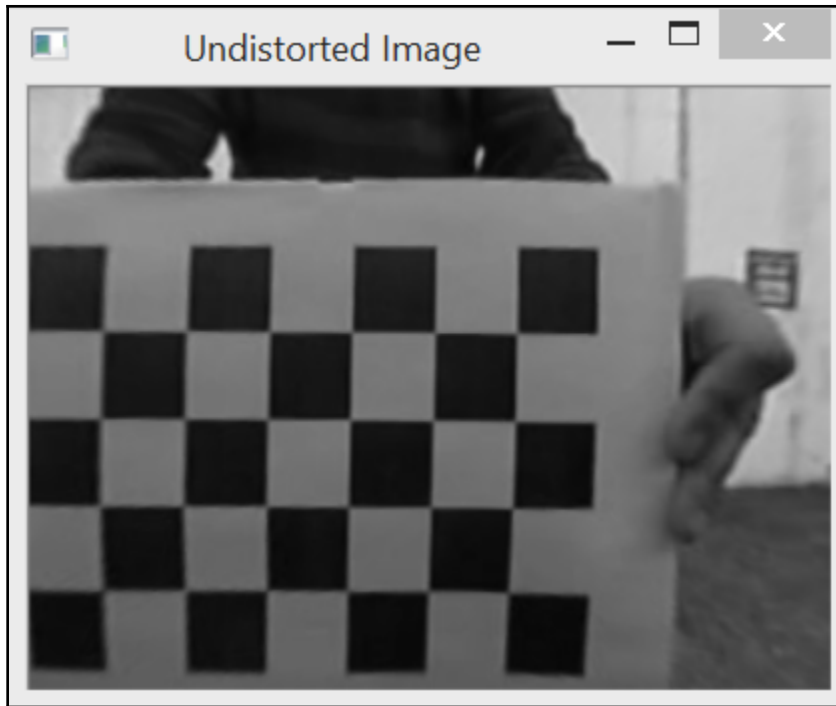


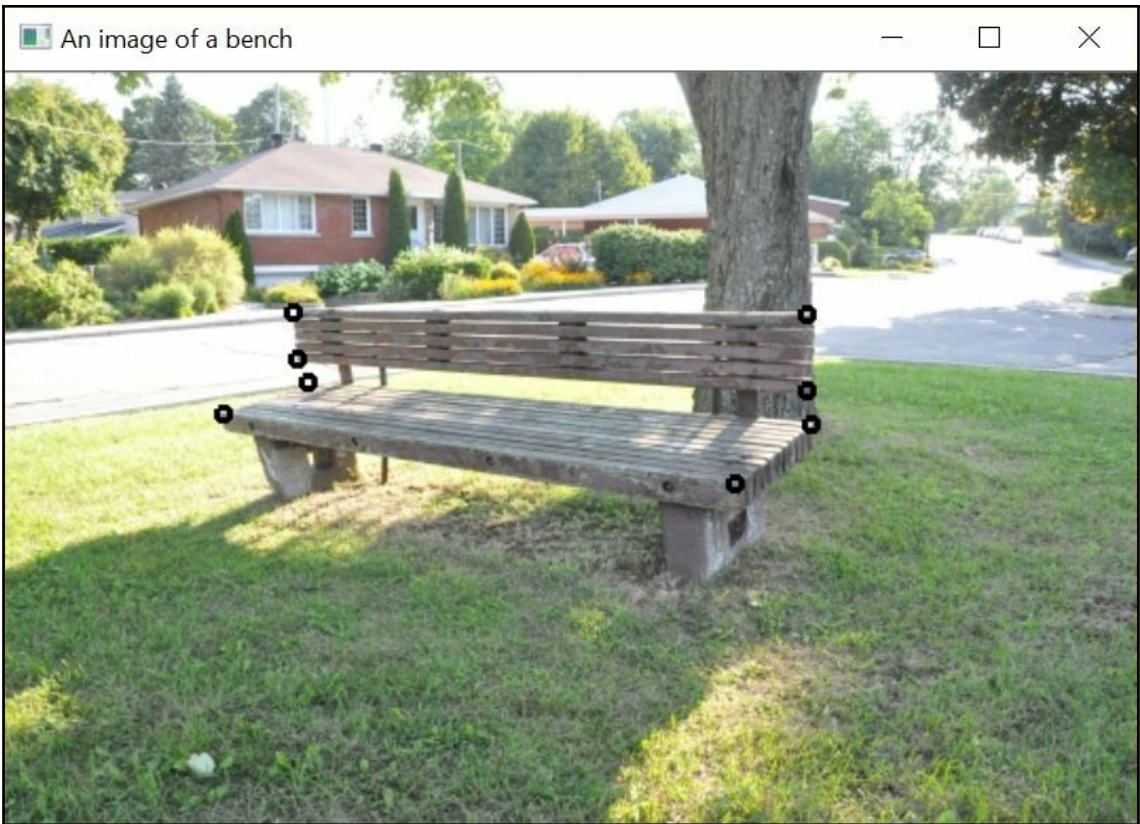


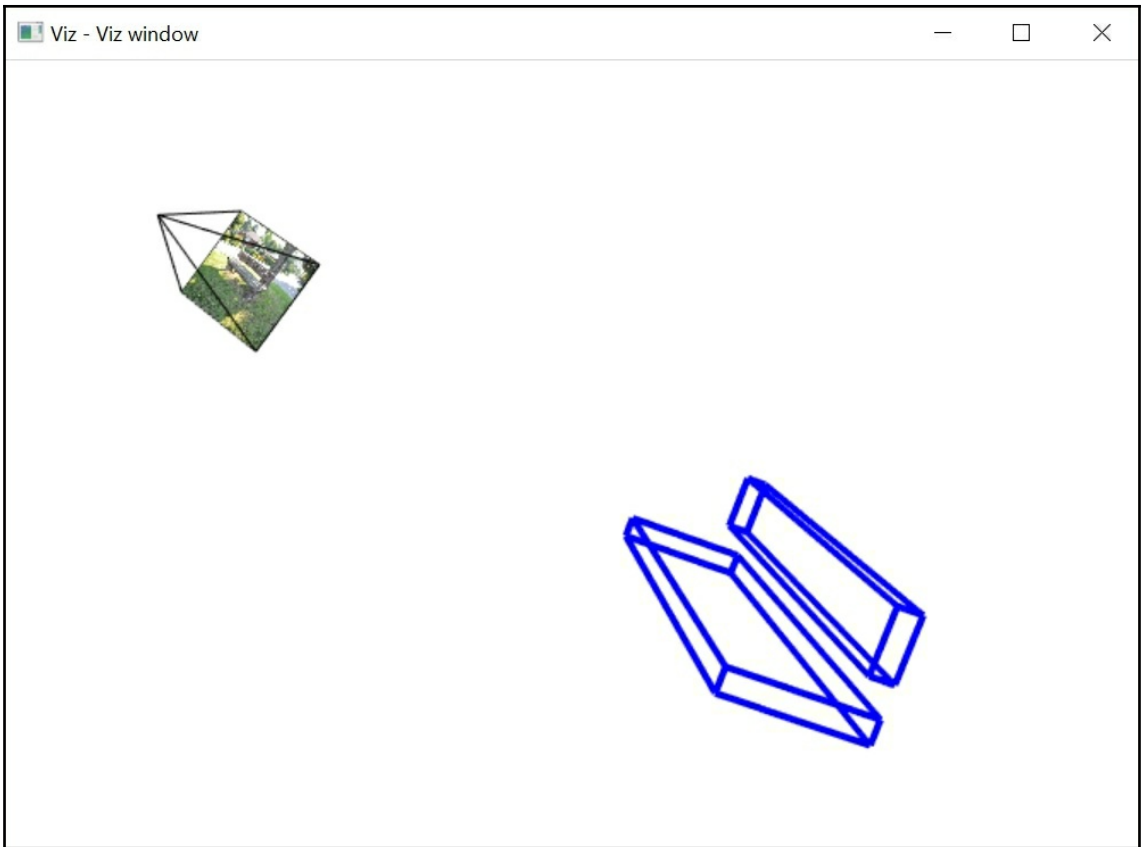
Chapter 11: Reconstructing 3D Scenes

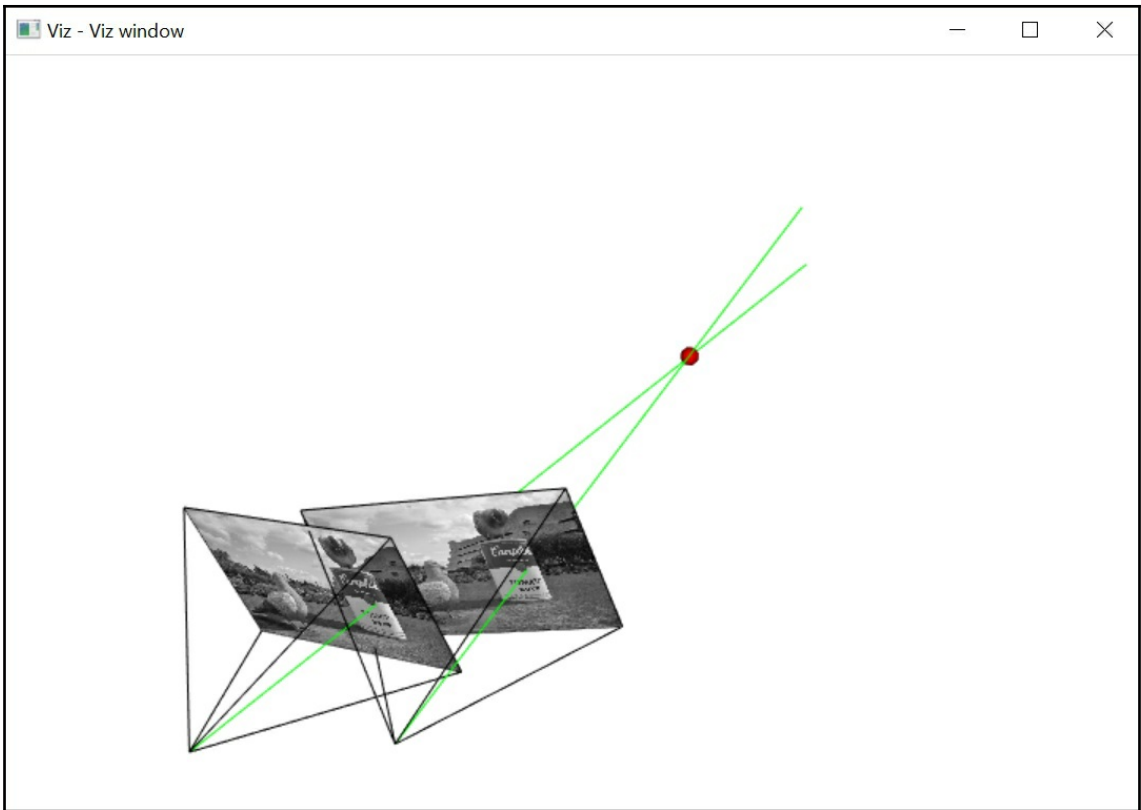


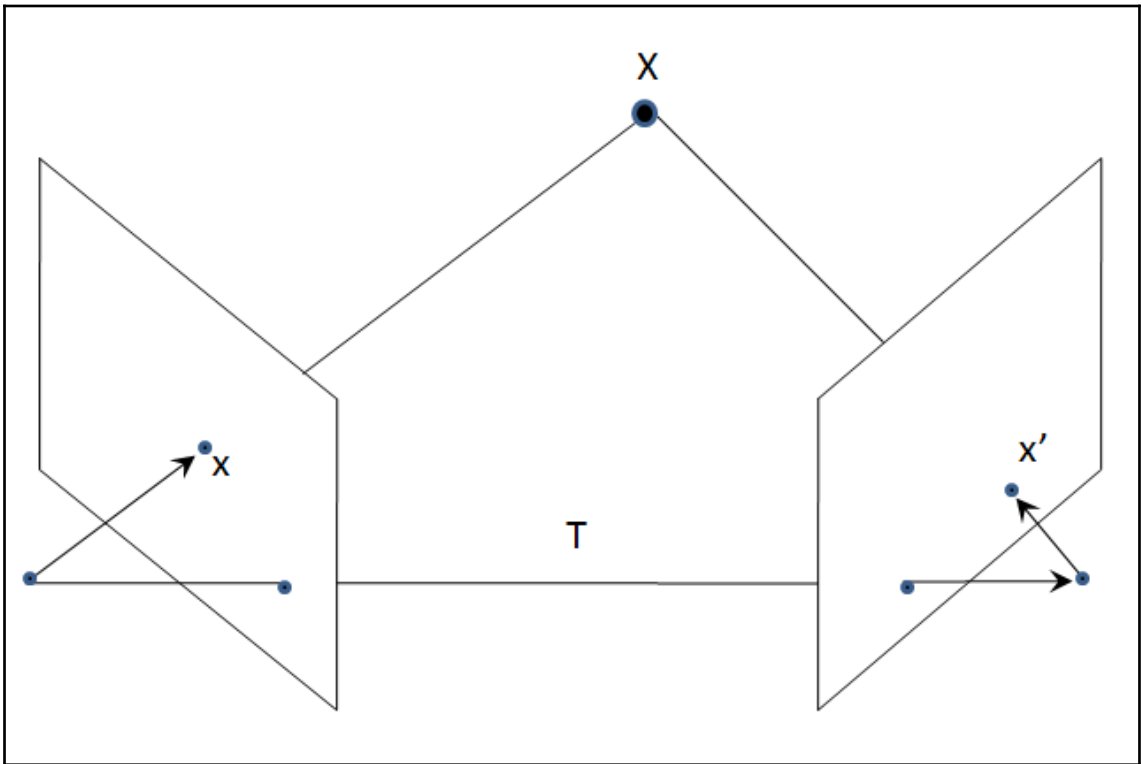
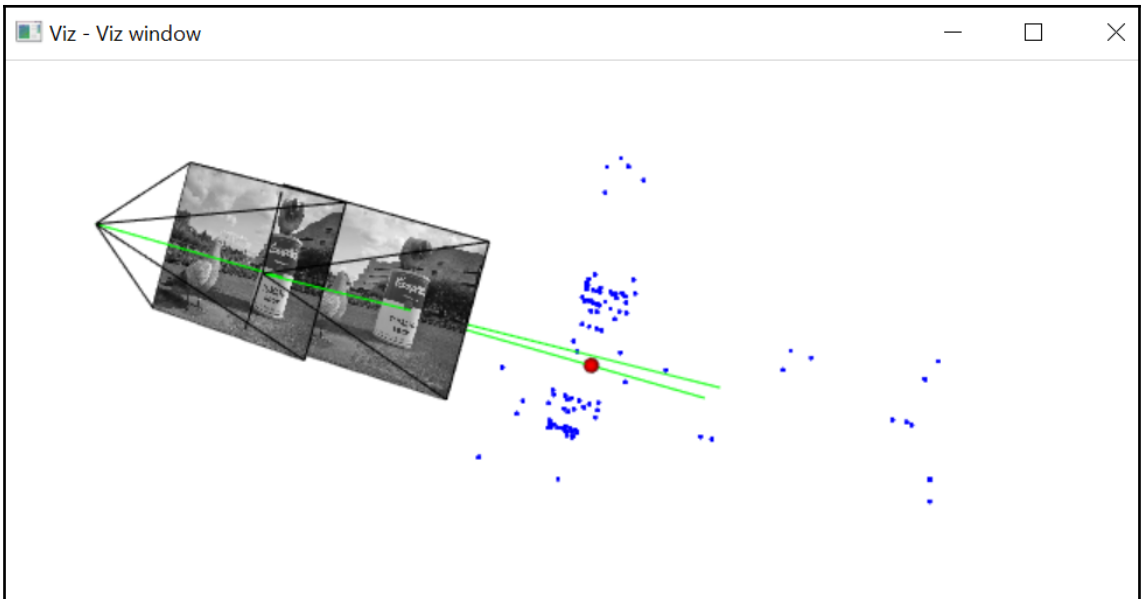


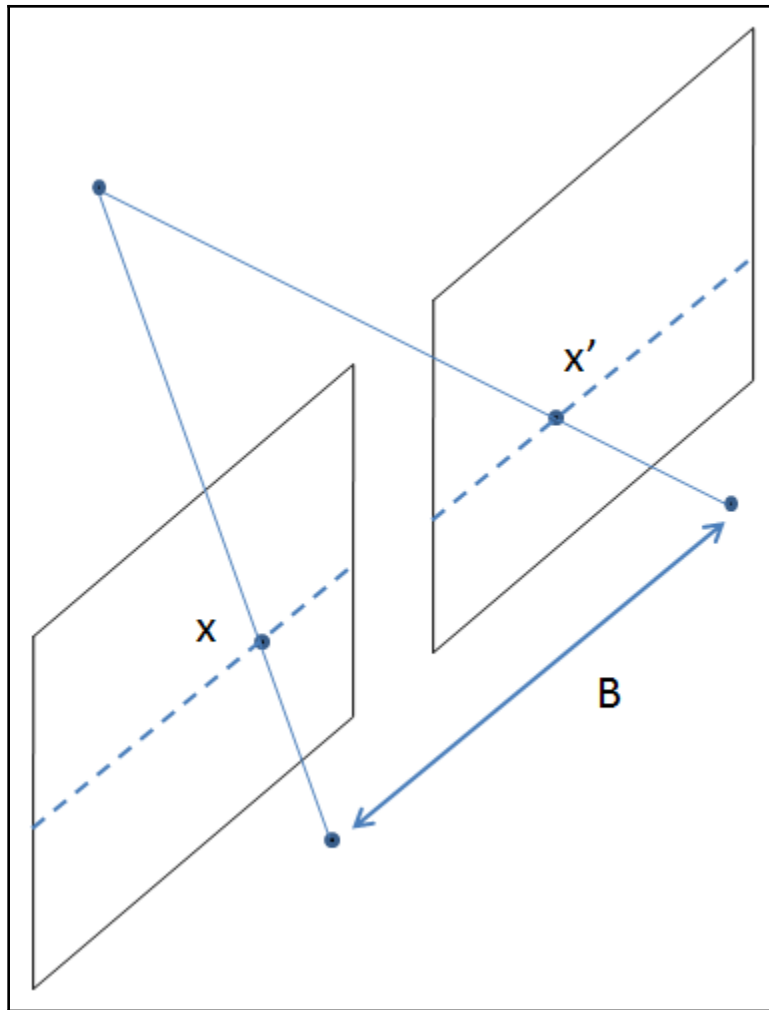




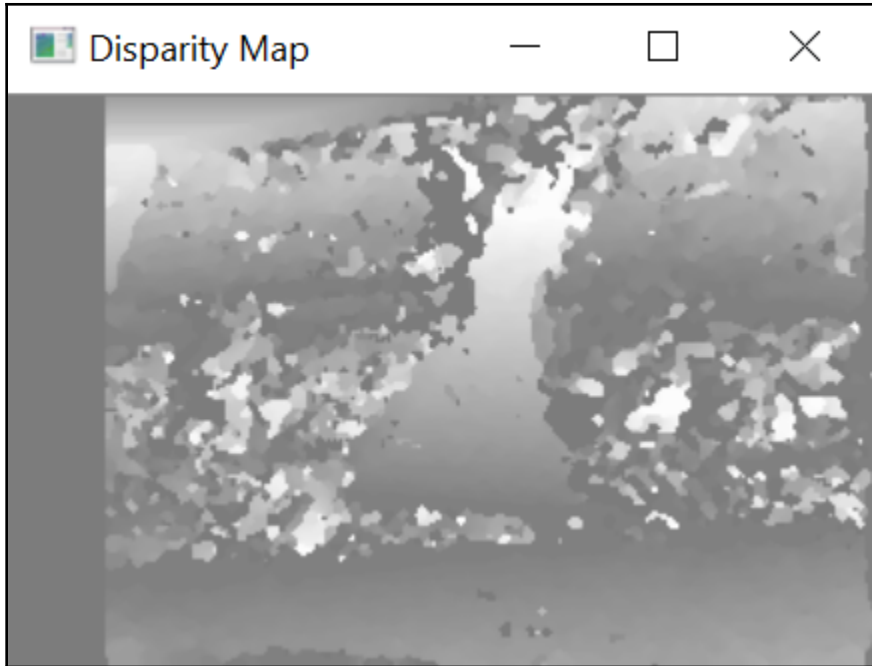




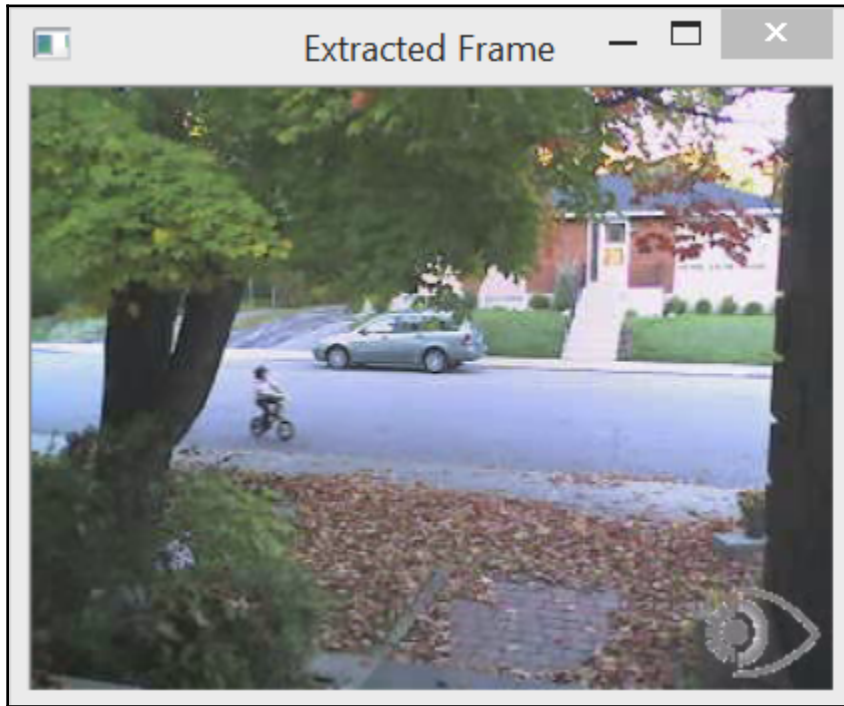


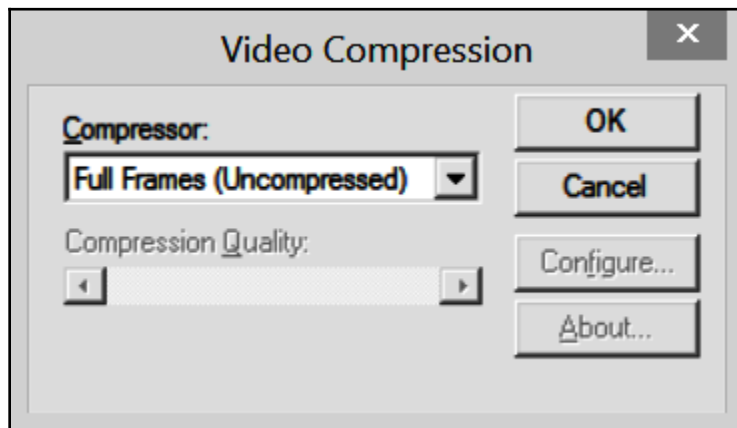
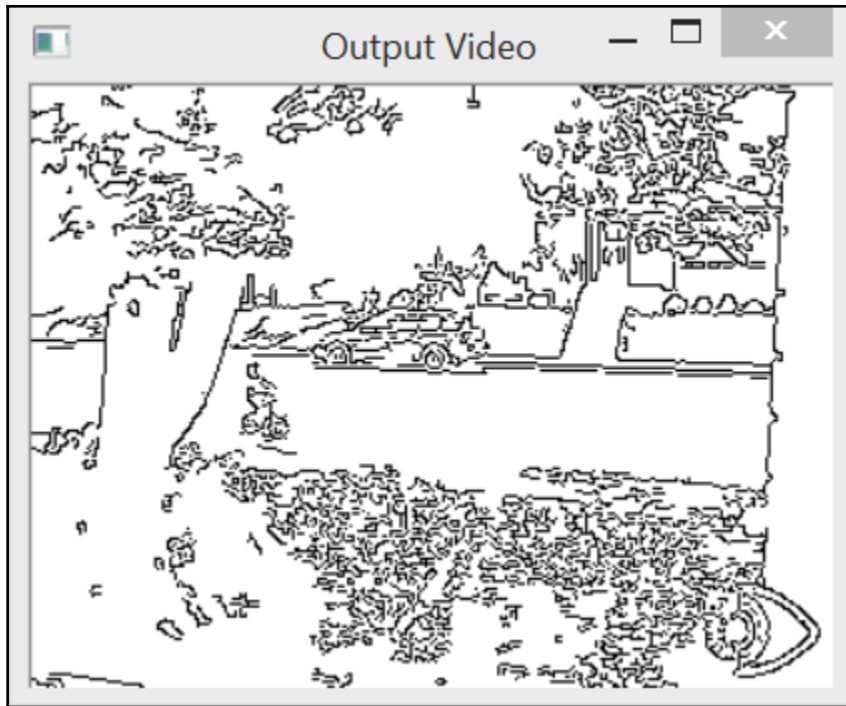


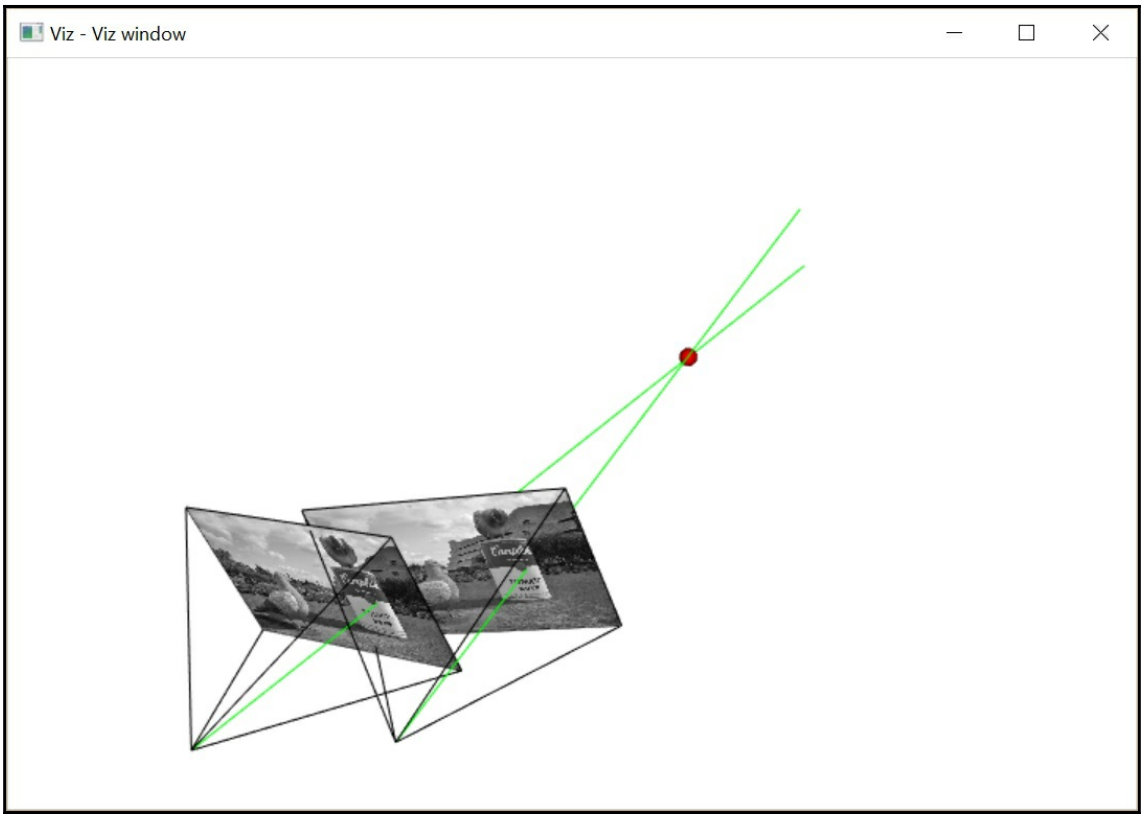




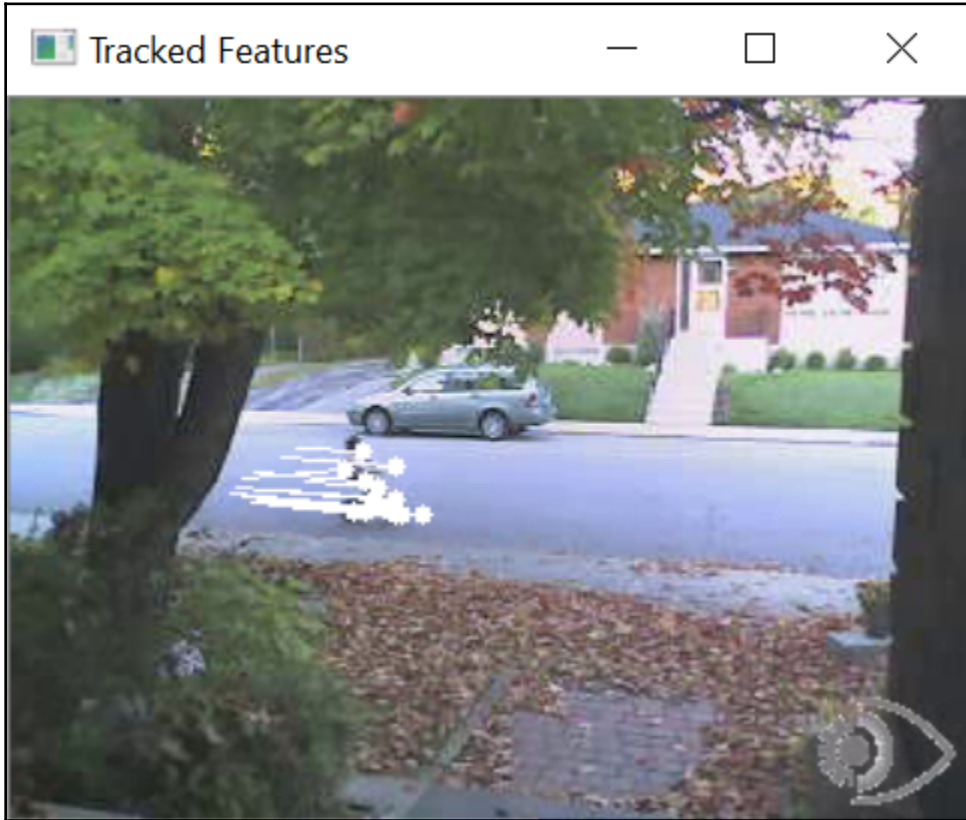
Chapter 12: Processing Video Sequences

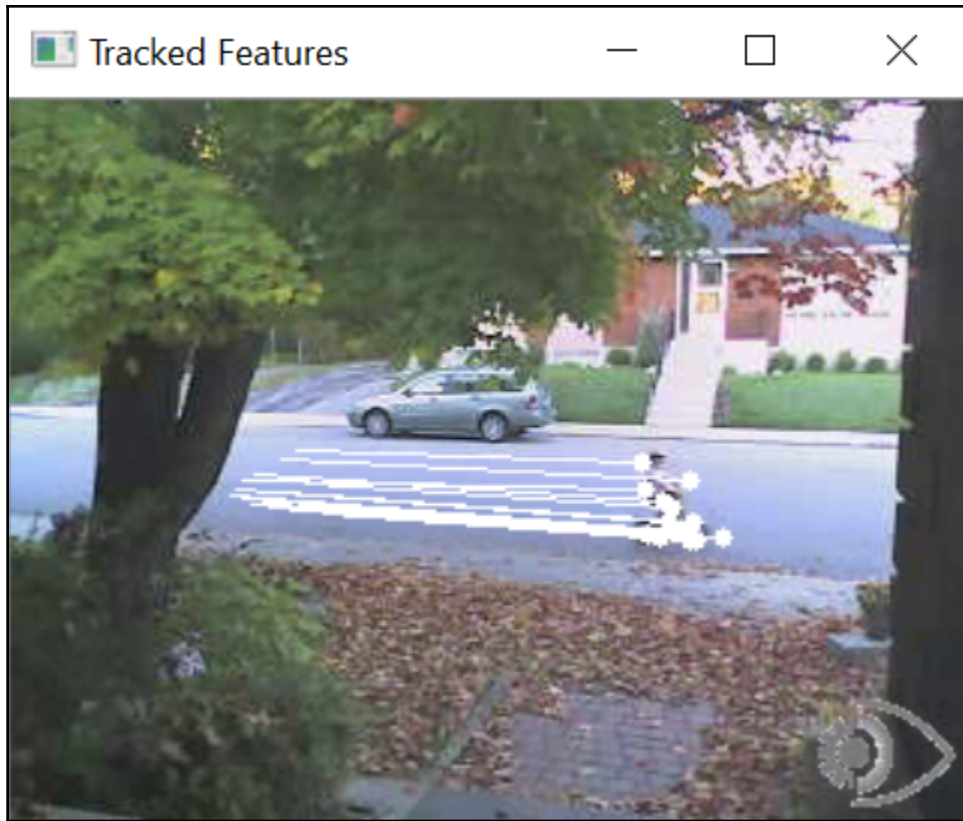


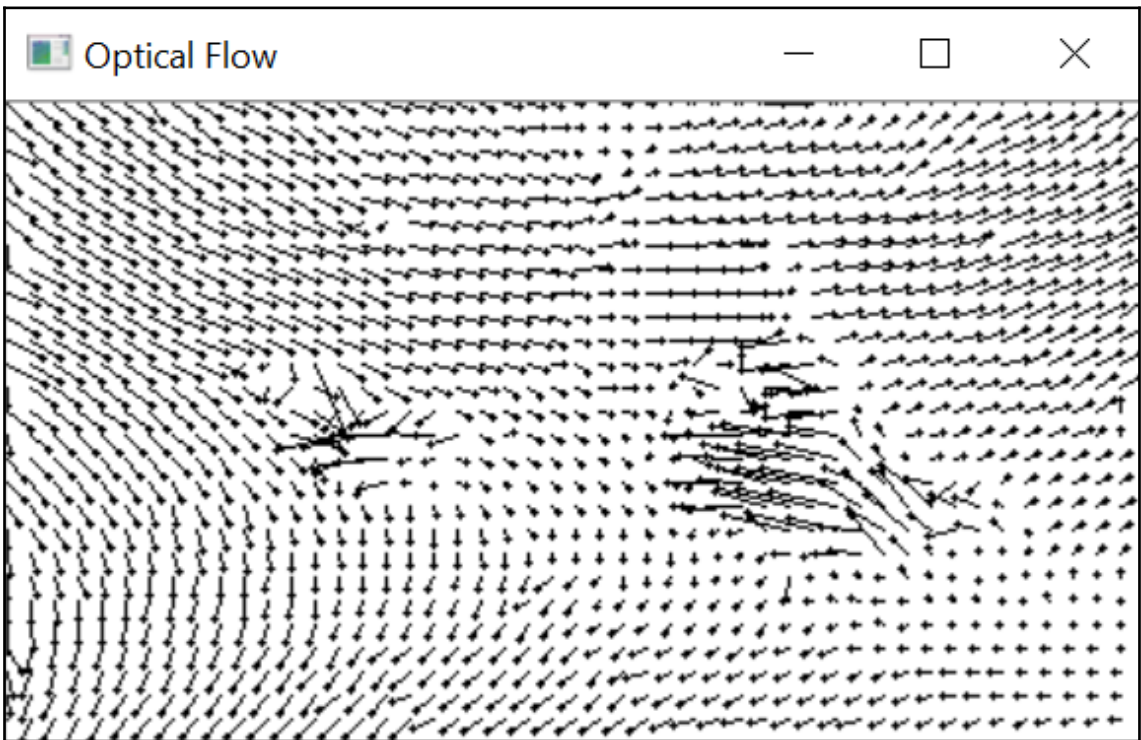
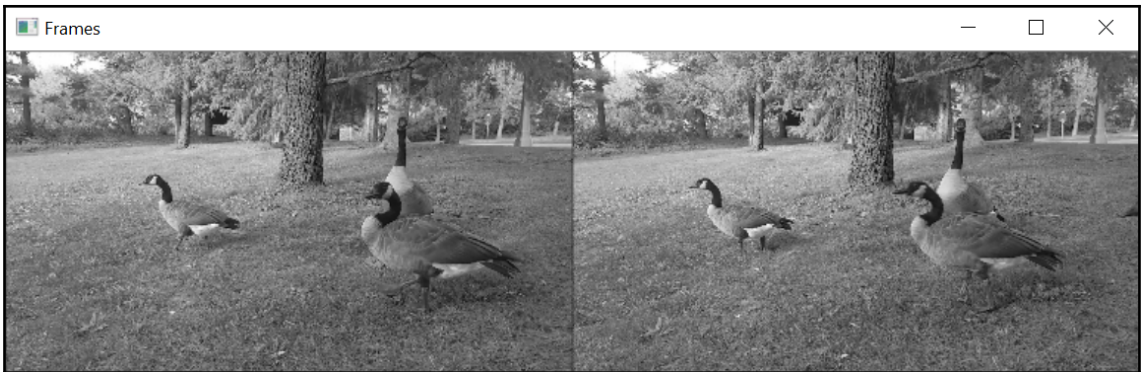


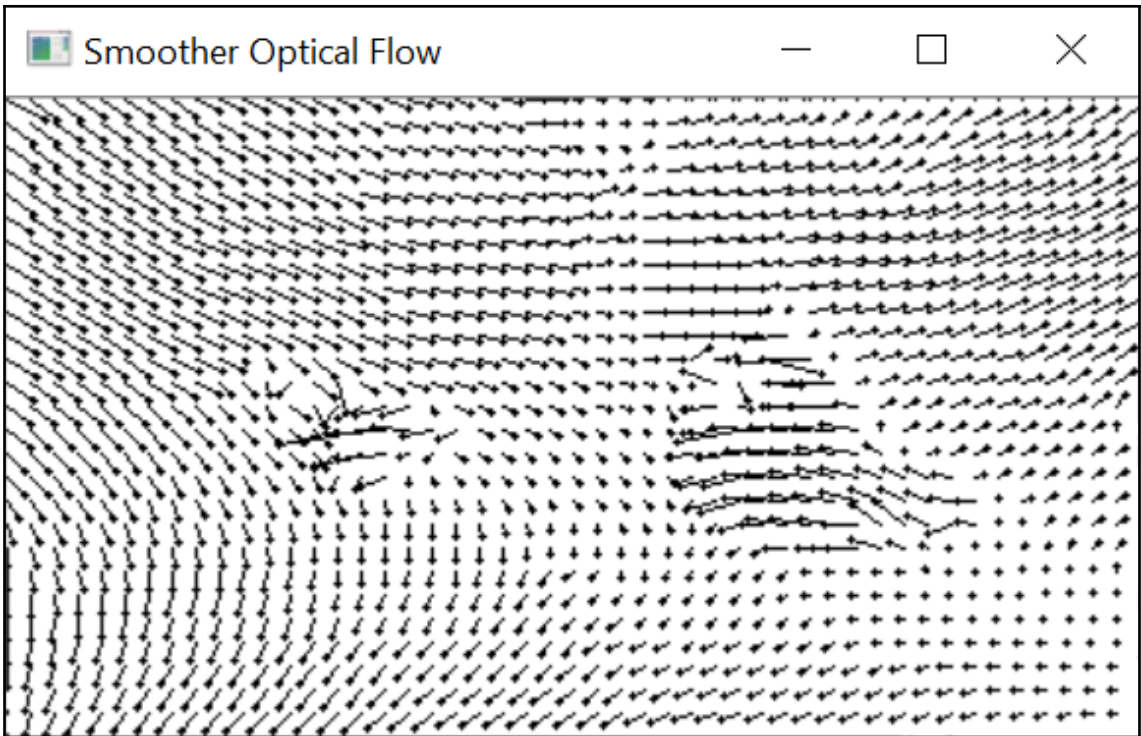


Chapter 13: Tracking Visual Motion

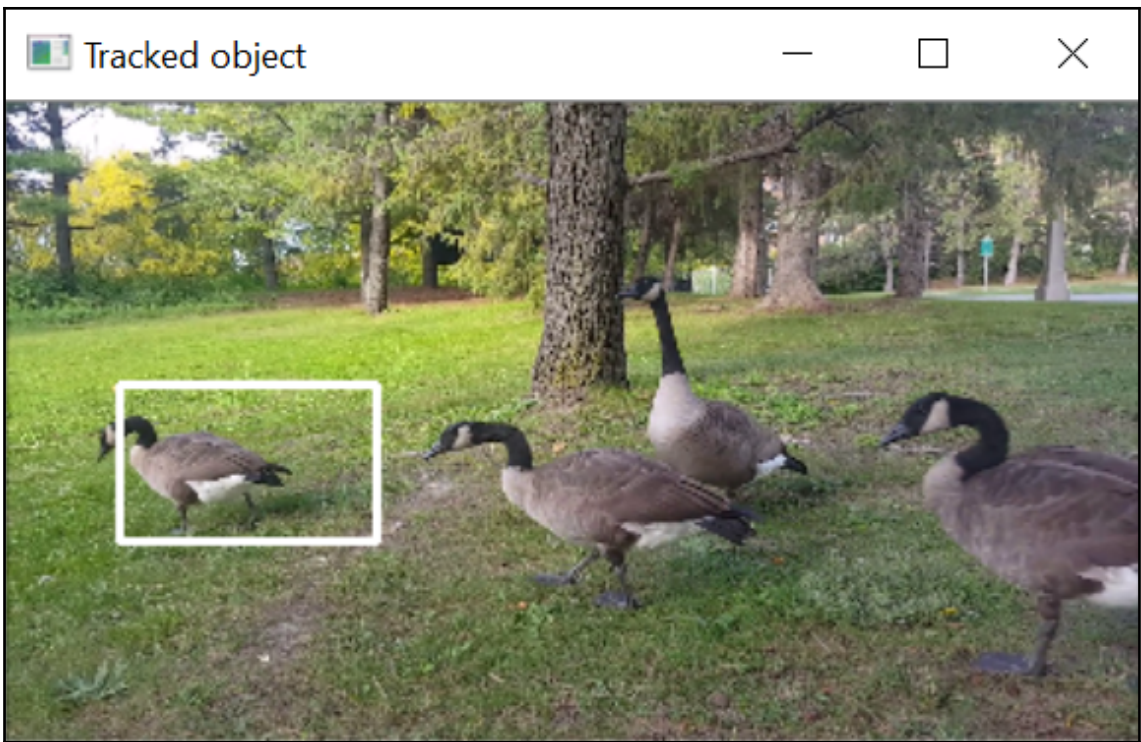




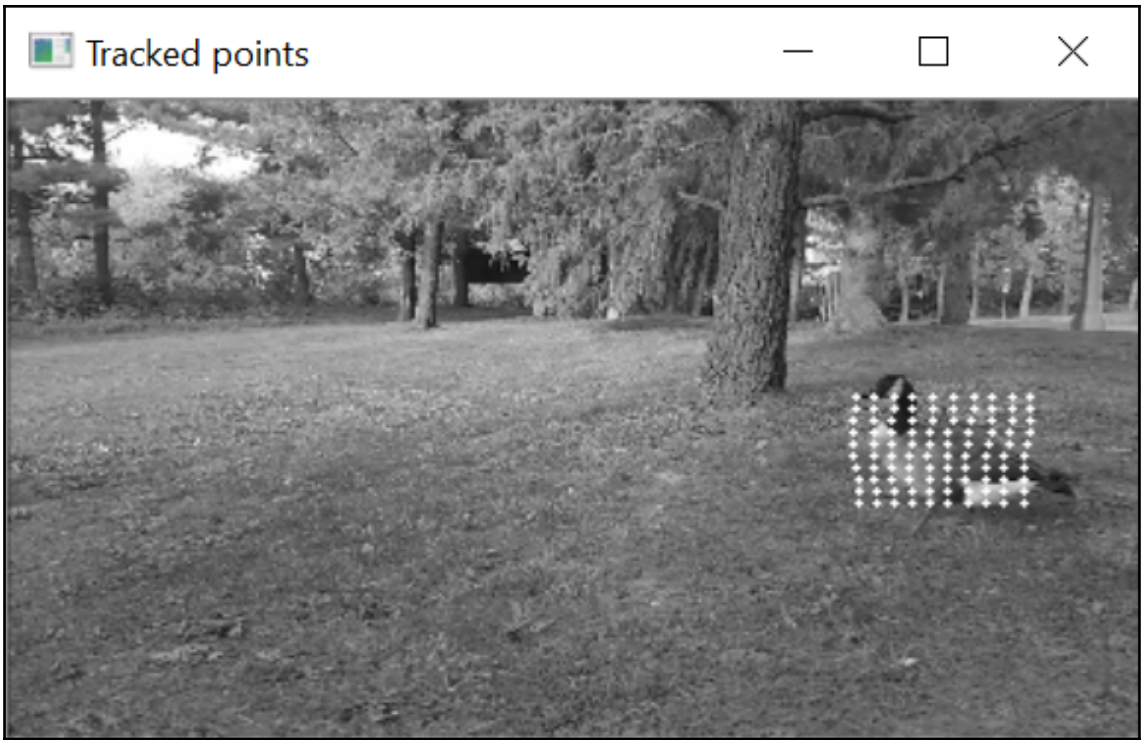


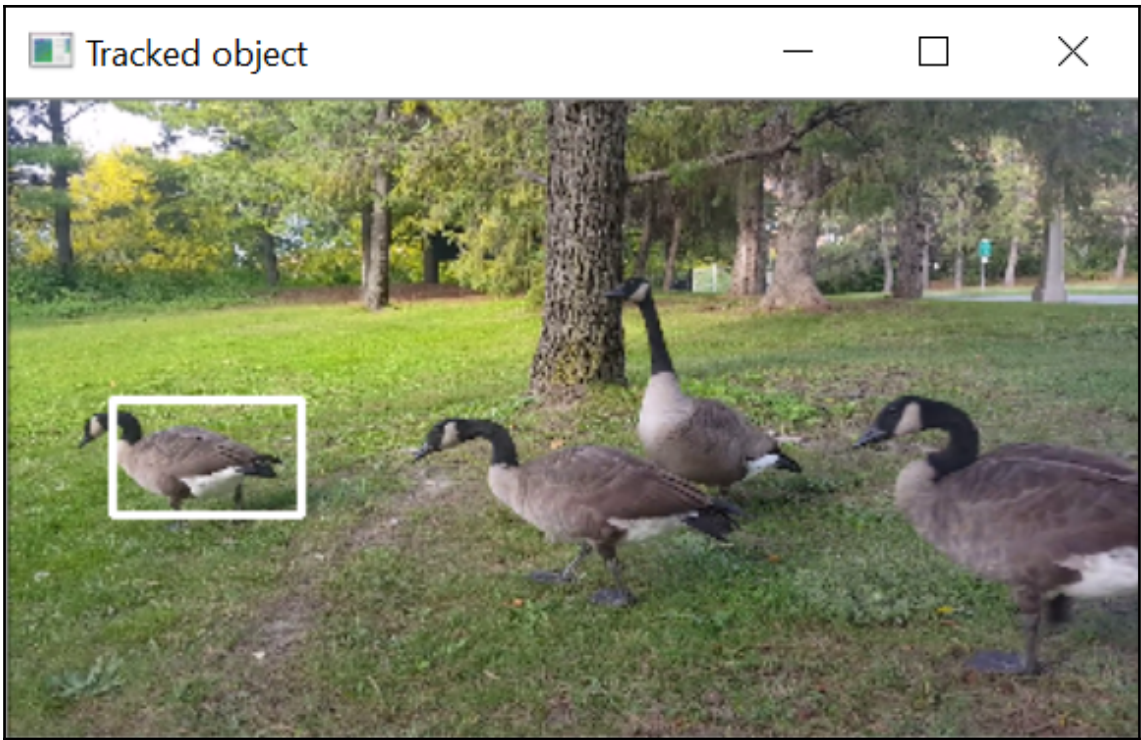




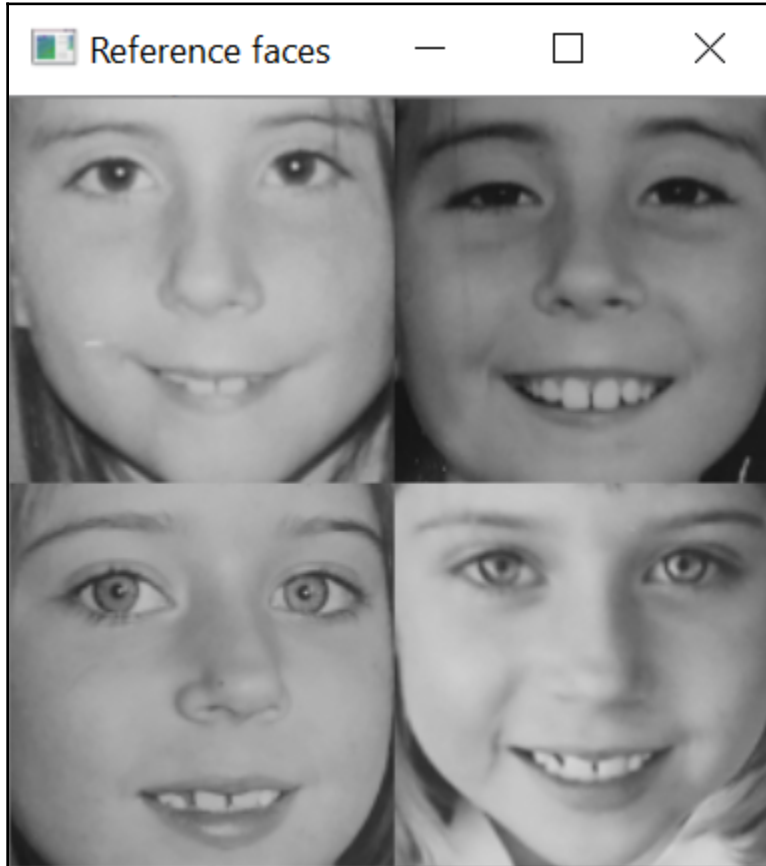


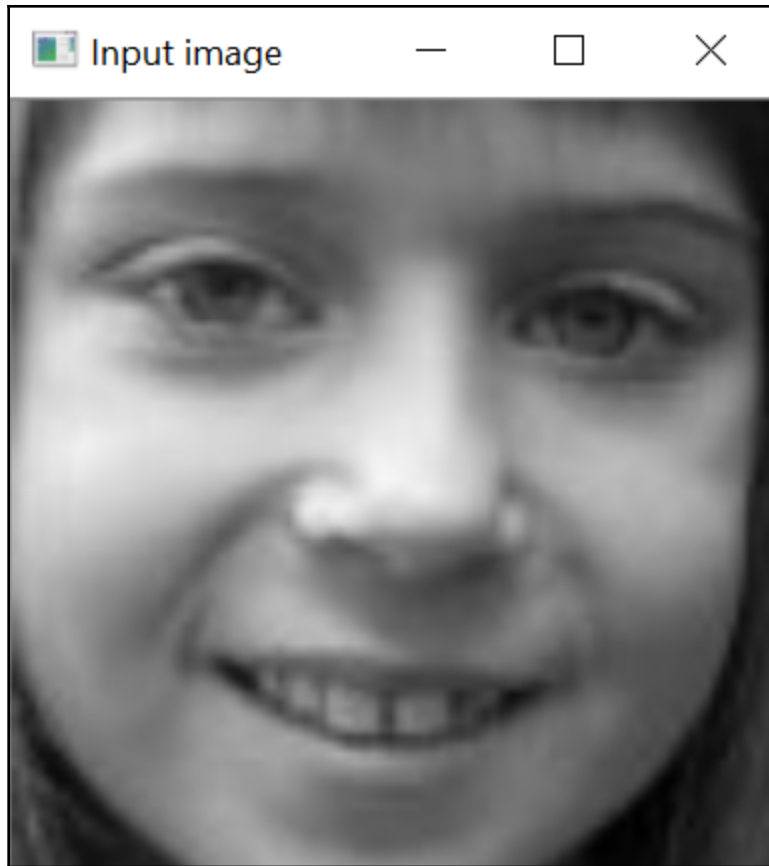




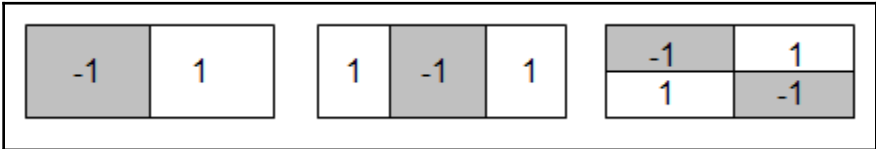


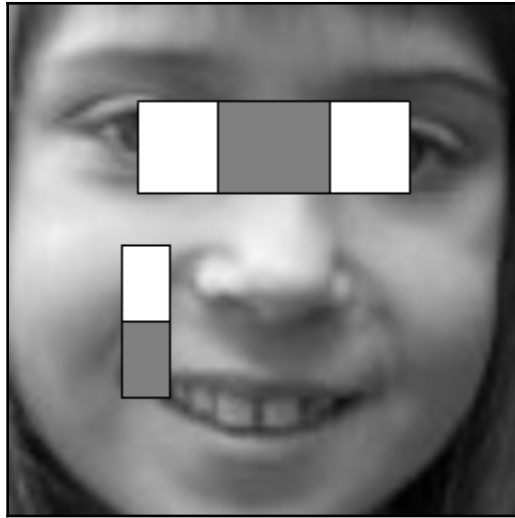
Chapter 14: Learning from Examples



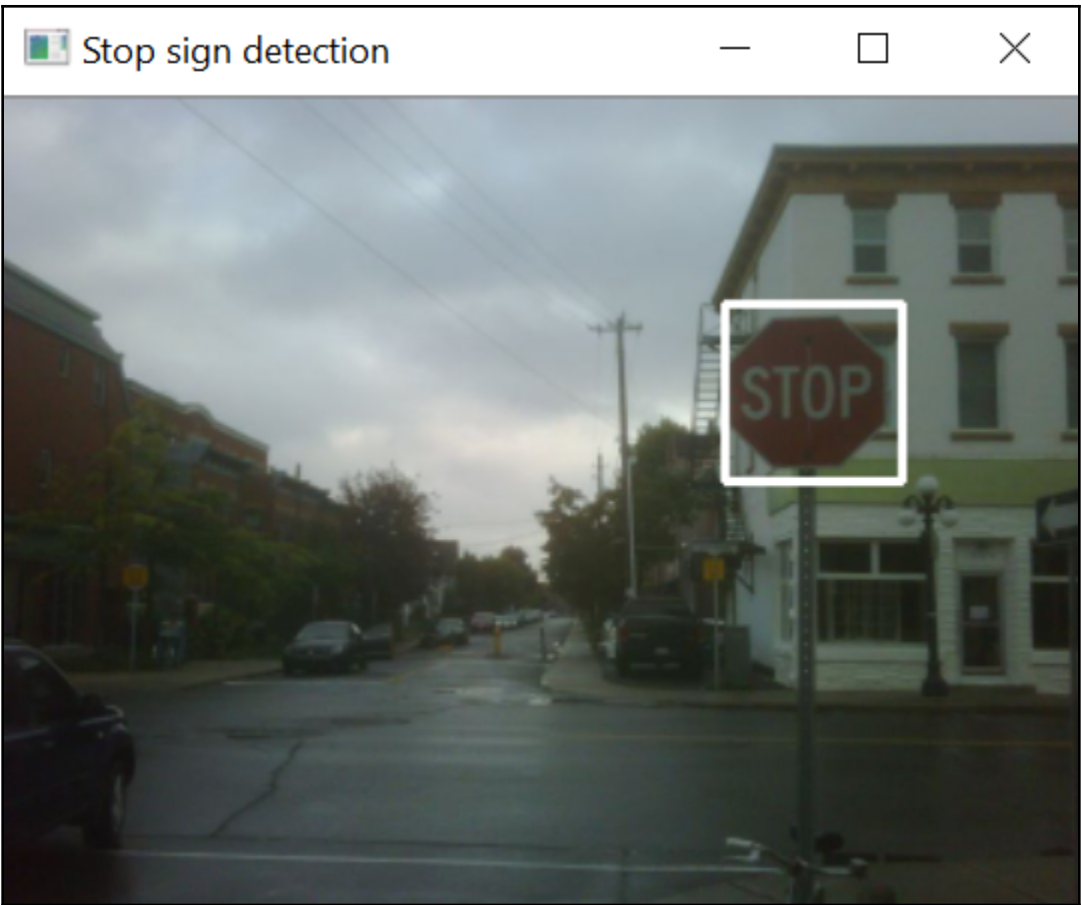


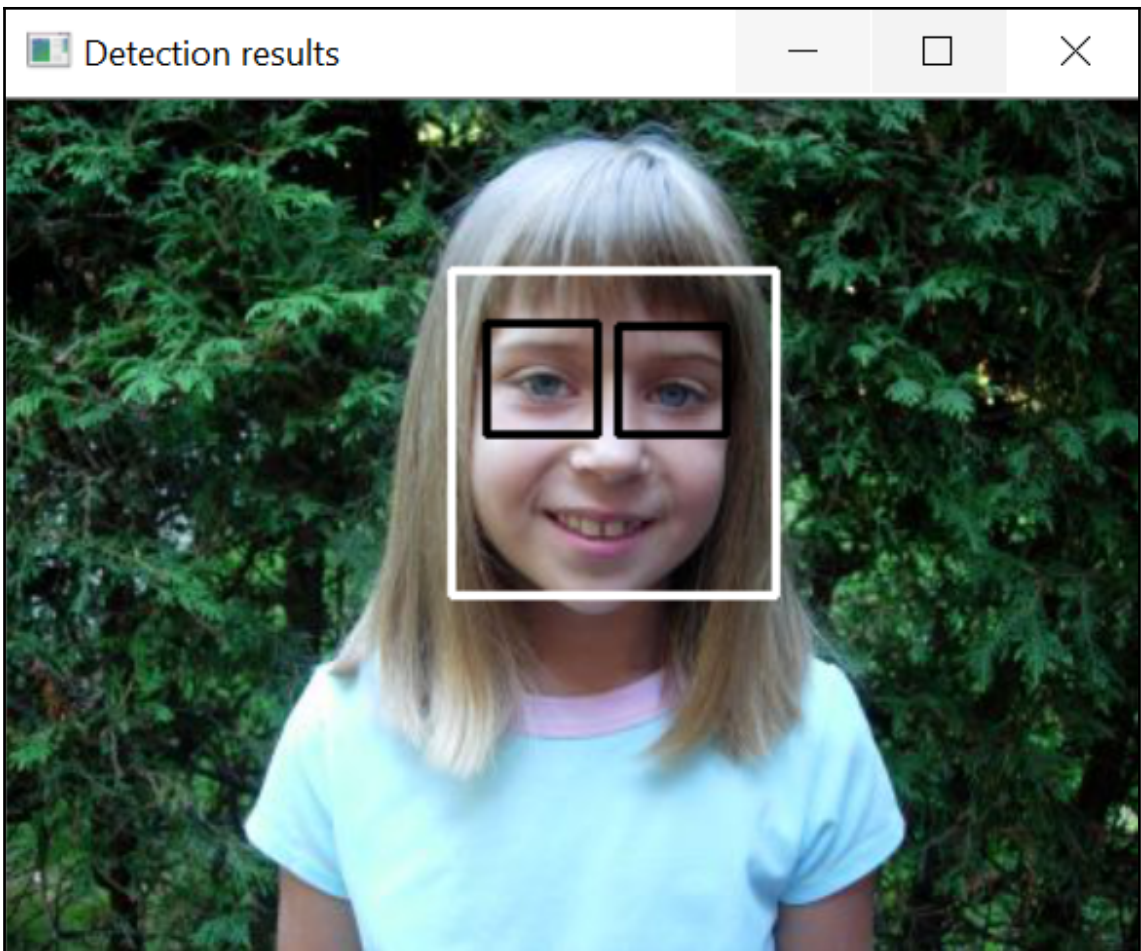




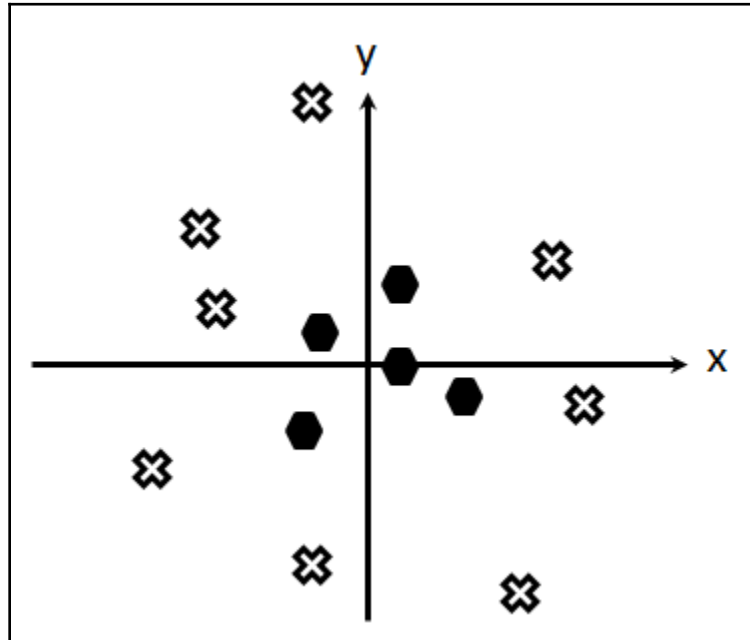
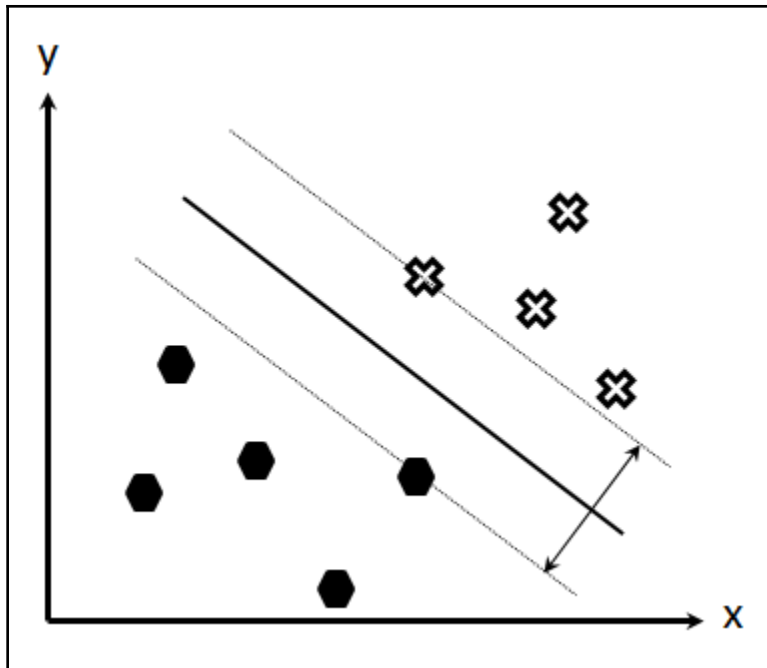


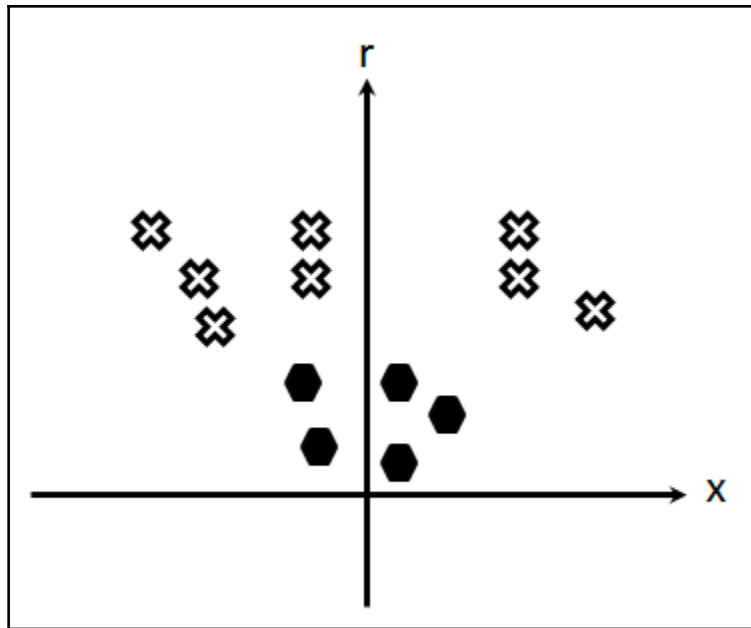


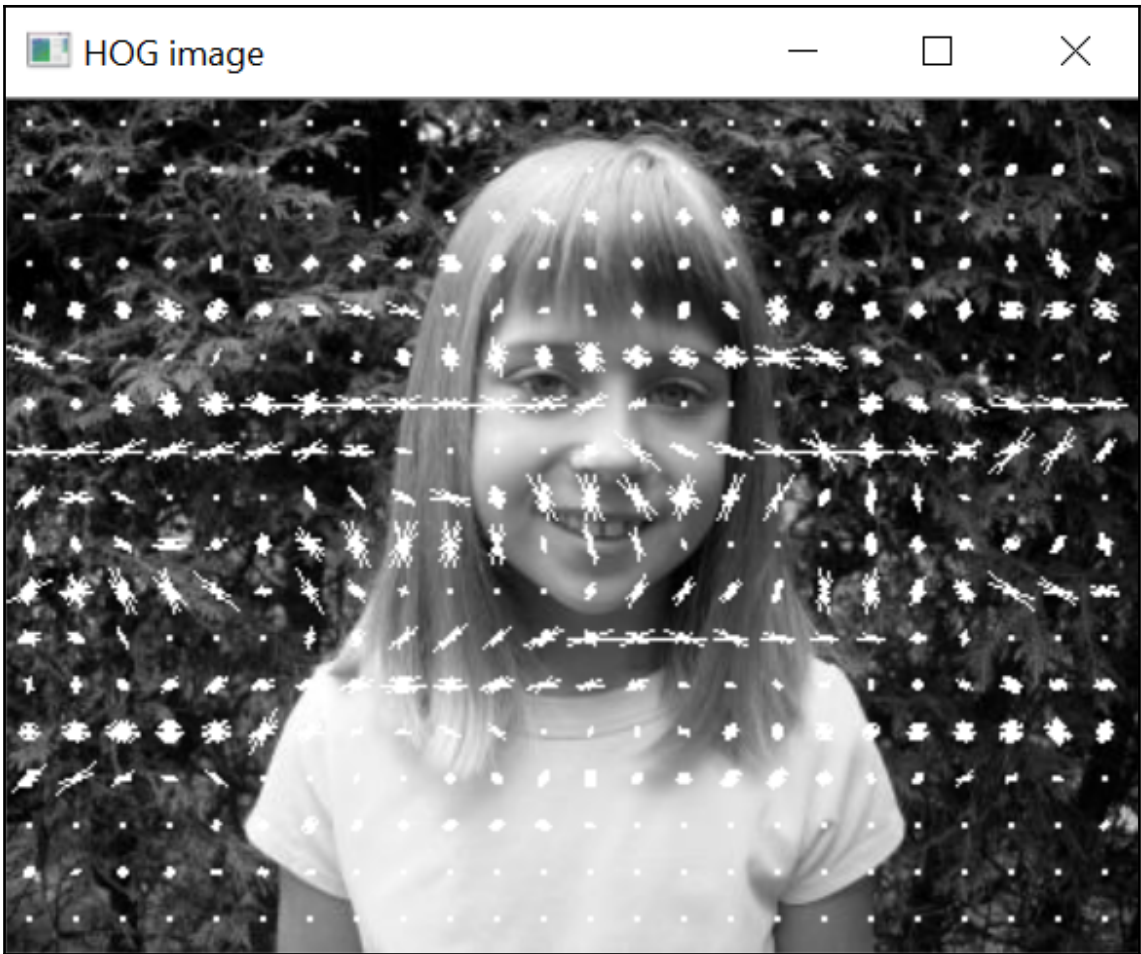








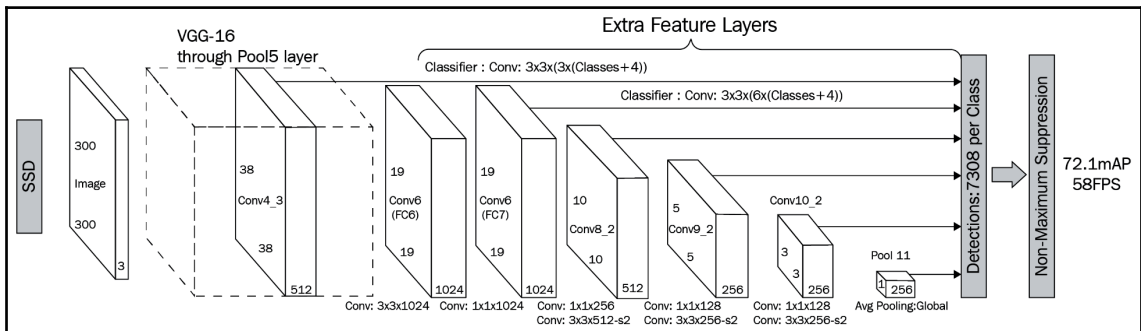
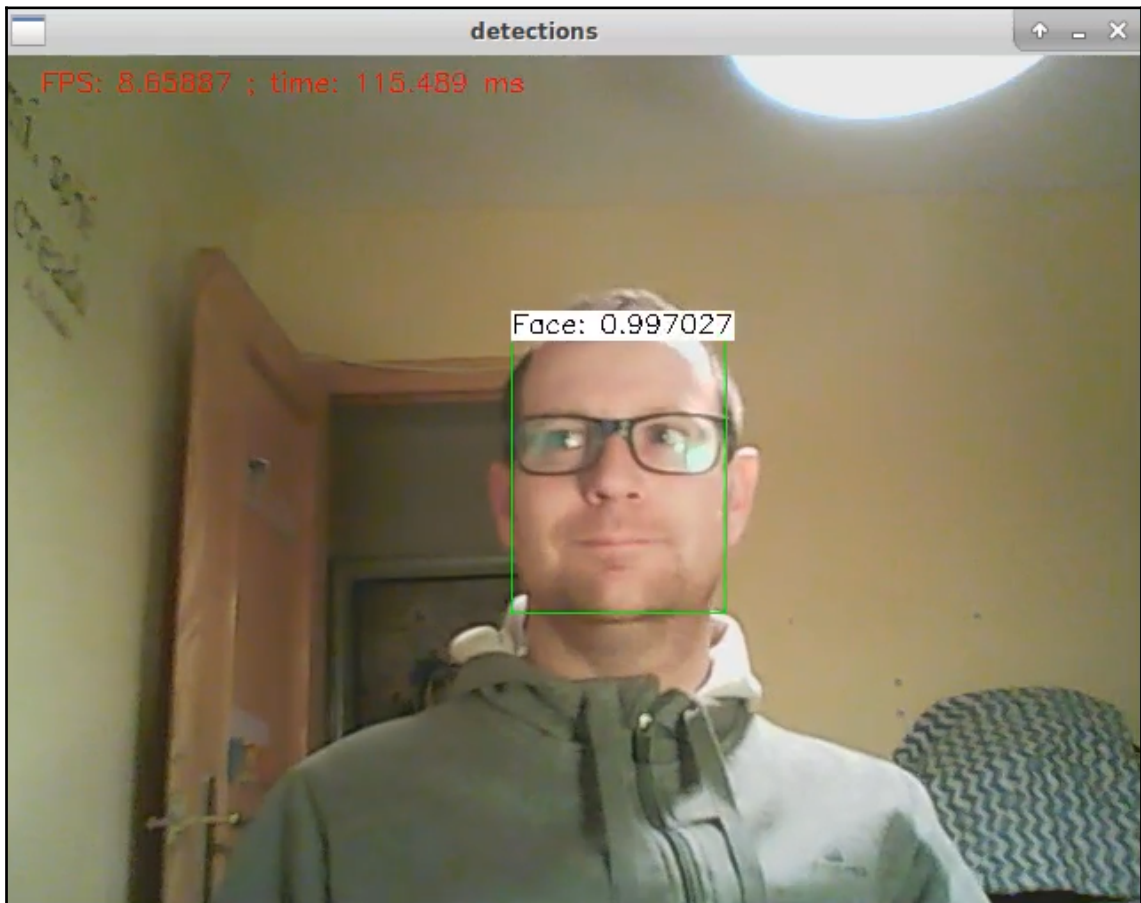


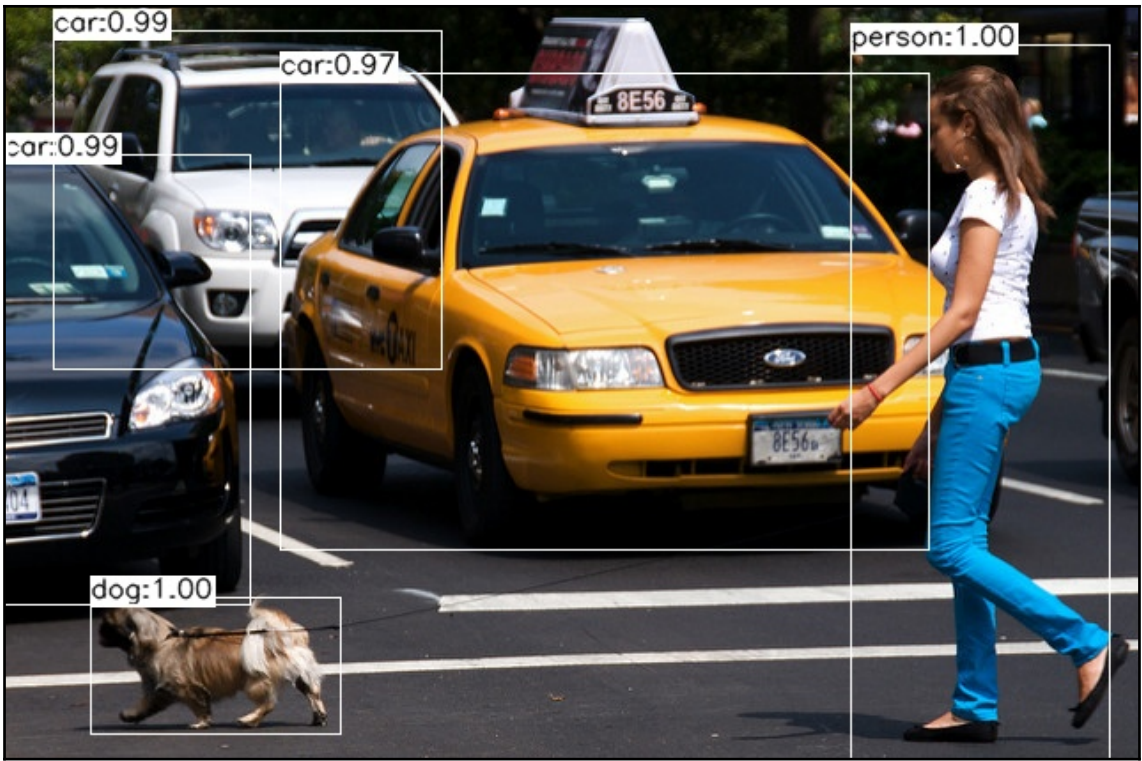


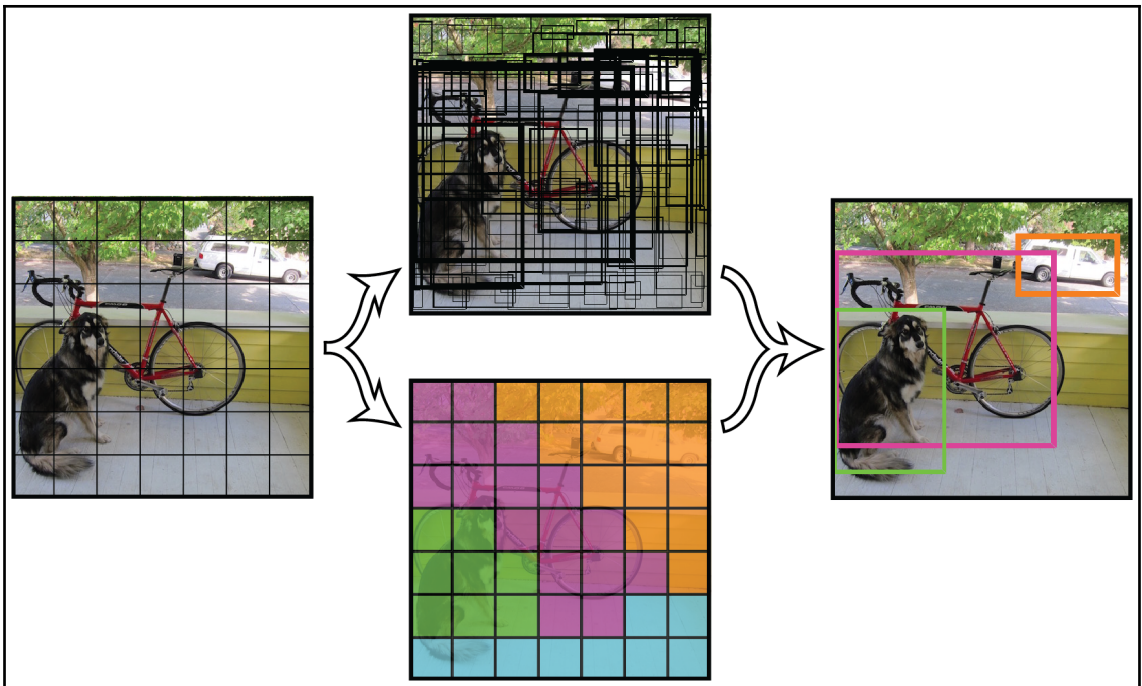


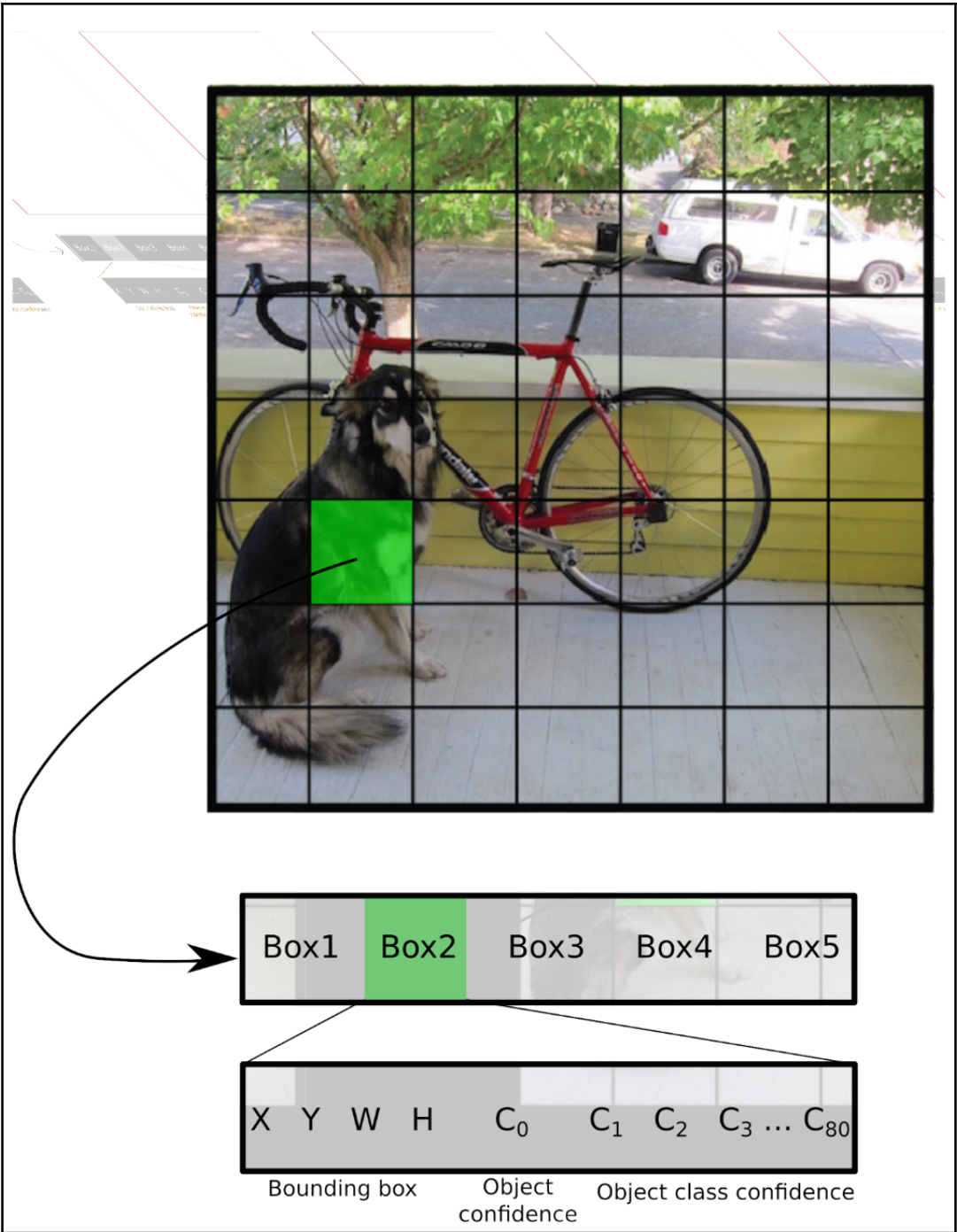


Chapter 15: OpenCV Advanced Features









OpenCV Computer vision on Wi x +

File | /home/damiles/Projects/OCV4-coockbook/OpenCV-4-Computer-Vision-Application-Programming-Cookbook/Chapter15/opencvjs_... ☆

OpenCV.js is ready. You can upload an image.
The `imageSrc` is a `` element used as `cv.Mat` input. The `canvasOutput` is a `<canvas>` element used as `cv.Mat` output.



Choose File lenna.png

4