Smoothing filters:
The output from the averaging filter that I implemented looks very similar to the output from the OpenCV averaging filter.
The output from the gaussian filter that i implemented also looks very similar to the output from the OpenCV gaussian filter, except in the one i implemented it leaves an outer layer of black pixels because of how the filter was implemented to deal with edge pixels.

These similarities occur because both averaging and both gaussian filters are using the same sizes, and both gaussians use the same sigmas.

The output of the sobel filter differs from mine, it has fewer lines and white throughout the image. This could be due to a thresholding that occurs after the smoothing happens, or maybe the calculations of pixel values are done at different times, because in my implementation we calculate the x and y pixel at the same time, and then find the magnitude of that, whereas sobel calculates the entire horizontal image, and entire vertical image, and then combines them

Edge detectors:
The marr-hildreth filter overall has thicker edges, more false positives, and more false negatives. The false positives can be seen by looking at the grass below the camera, and the false negatives can be seen in the buildings in the background. I think the algorithms themselves are what contribute to these differences the most, as I tried many variations of zero crossing thresholds to get a better marr-hildreth image, and still couldn't get as good of an image as I got from Canny.

Edge map:
The Canny edge map resulted in more connected components, which can be seen very clearly in the cameraman's face and torso. This is a result of having more connected edges overall, and cleaner edges with less noise and thickness. The Canny edge map had the best result where each coloured region is typically a part of an object in an image, and not connected to other areas.