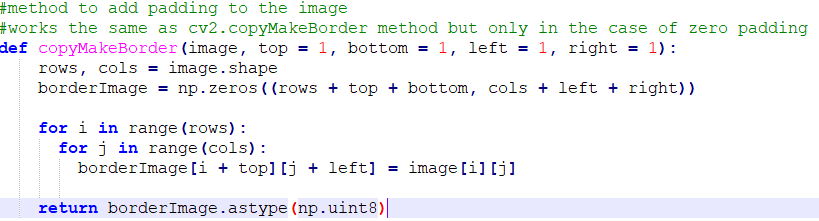
**COMP37212 Computer Vision**

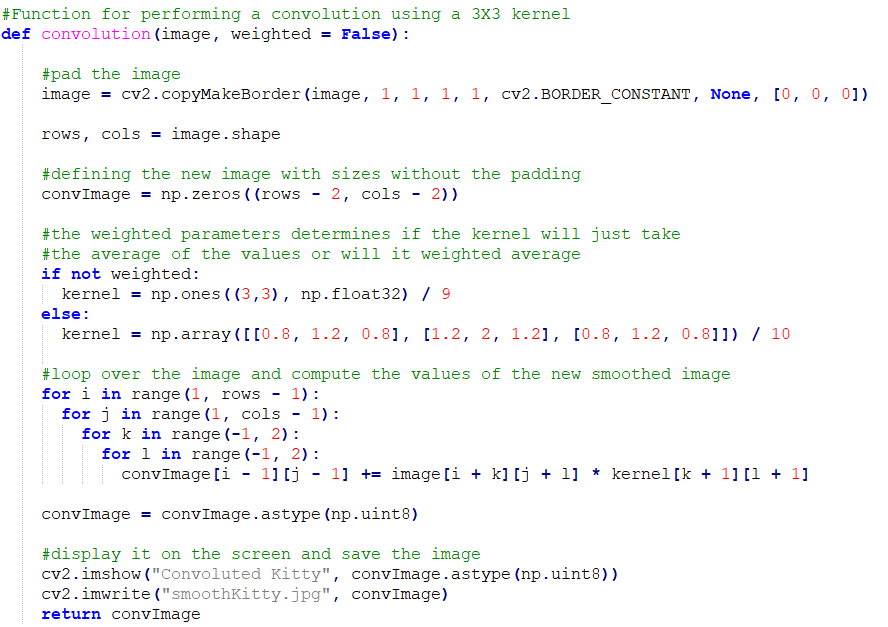
**Convolution & Kernels**

1. Python code

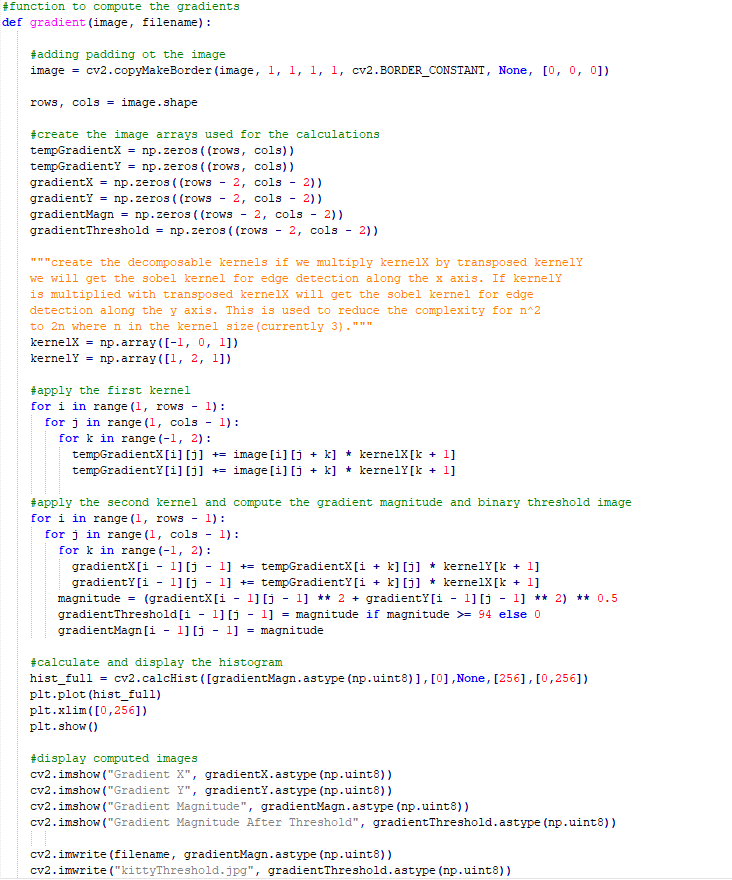
* Padding function



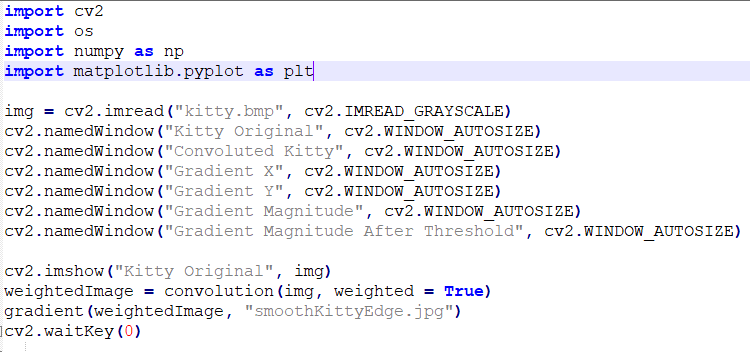
* Convolution function



* Gradient function



* Rest of the code



1. Convolution

* With mean kernel



* With weighted-average smoothing kernels



1. Horizontal Gradient Image



1. Vertical Gradient Image

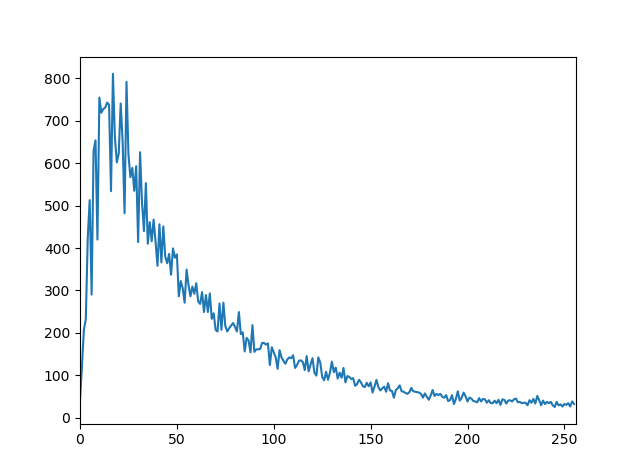


1. Gradient Magnitude



1. Histogram

Histogram of the image before any threshold is applied



1. Threshold

When applying a binary shreshold of 135.



1. Smoothing Kernel

Vertical and horizontal gradient images with smoothing kernel

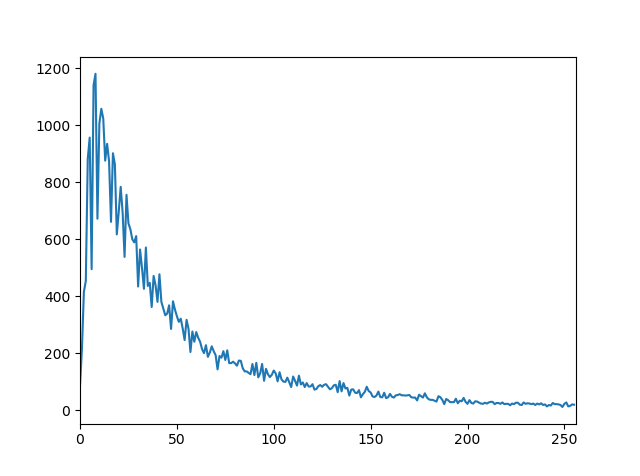




Gradient magnitude of smoothed image



Histogram of the gradient magnitude of the smoothed image



When applying the smoothing kernel beforehand the binary threshold to produce the image is decreased to 94 which matched the suggested value by otsu thresholding.



1. Discussion
   1. *Can you find a threshold value that gives the edges of the cat, but not the patterns in the fur, or the wood grain?*

*No such a value does not exist. If we increase the thresholds the wood grain will be filtered out but the cat’s head gets filtered out before the patterns in the fur which ruins the edges of the cat*

* 1. *Comparison of the edge-strength images/What difference has the weighted-mean smoothing made to the edges detected?*

When compared the edge strength images before any threshold is applied there is a clear difference. Due to the smoothing convolution applied some of the small edges in the wood grain, the ball of yarn and the kitty’s face are not detected anymore. This will lead to a lot less unnecessary information and much cleared edges of the cat’s body.

* 1. *Threshold value*

*The value was chosen to have filter out as much of the wood grain and patterns on the fur as possible. At some point the cat’s edges started losing their shape so the threshold could not be increased more. Also when a weighted-average smoothing kernel was used beforehand the required threshold value decreases significantly.*

* 1. *Choice of weighted kernel*

The used weighted-average kernel is:

0.08 0.12 0.08

0.12 0.20 0.12

0.08 0.12 0.08

It was chosen because it has similar properties to a Gaussian kernel. The closer the value is to the center of the kernel the bigger the weight applied while at the same time the difference between the weights is not too big because that would result in a kernel which does not filter noise out well enough.