## Xiaotong Pseudo-code for 4750

#Other

def collect(images, scale, filters, window, overlap, border):

Initialize features and number of features

**FOR** all image in images **DO**:

Extract the features of image by calling F = extract(image, scale, filters, window, overlap, borader)

Update number of features and record the features F

Check if feature size of F is consistant

**END FOR**

Combine all the features and adjust them according to the offset

return the resulting features

def extract(image, scale, filters, window, overlap, border):

Initialize the grid by calling grid = sampling\_grid(X.shape, window, overlap, border, scale)

Set feature size to be window[0] \* window[1] \* len(filters) \* scale \* scale

if no filter is defined:

Use grid to extract features from image

else:

**for** each filter in the filters:

Use grid to extract features from the filtered image

Combine the features together

**end for**

return the resulting features

def sampling\_grid(imgSize, window, overlap, border, scale):

Resize the window, overlap and boarder according to the scale

Create sampling grid for overlapping window

Adjust sampling grid according to the offset size

return the resulting grid