CS484 Computer Vision

Ahmet Tuna Baykal

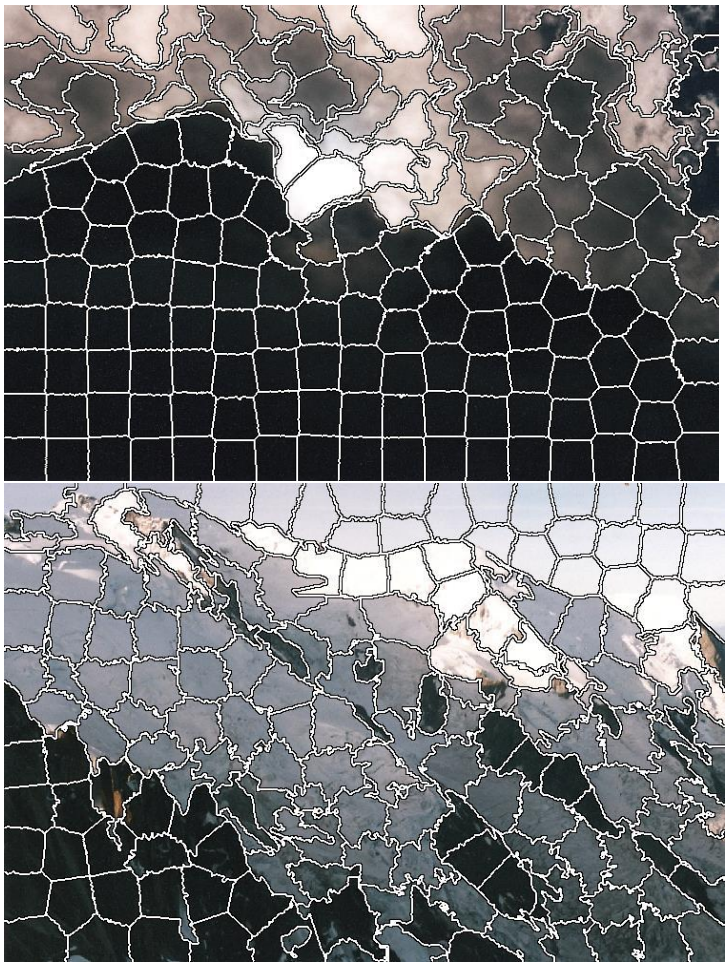
21803279

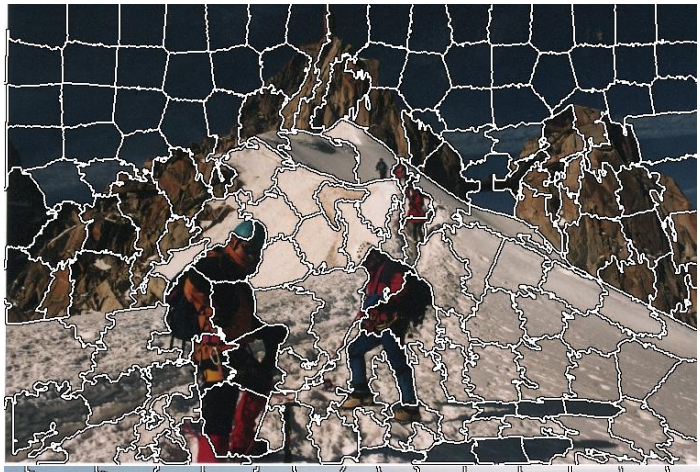
HW3

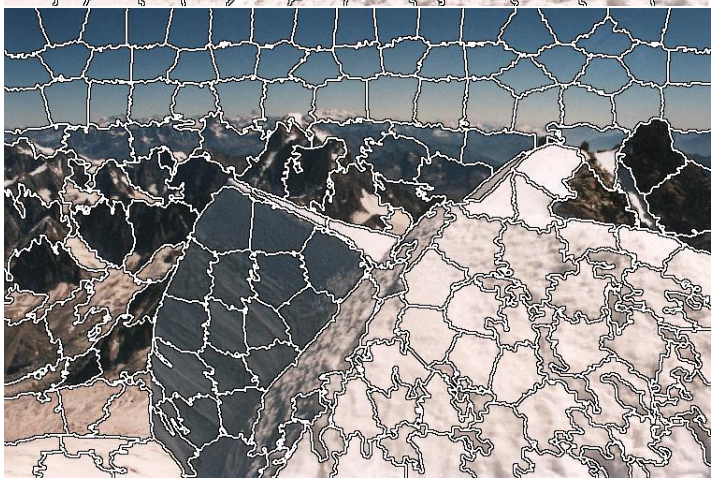
1) Description of the parameters used for superpixel segmentation.

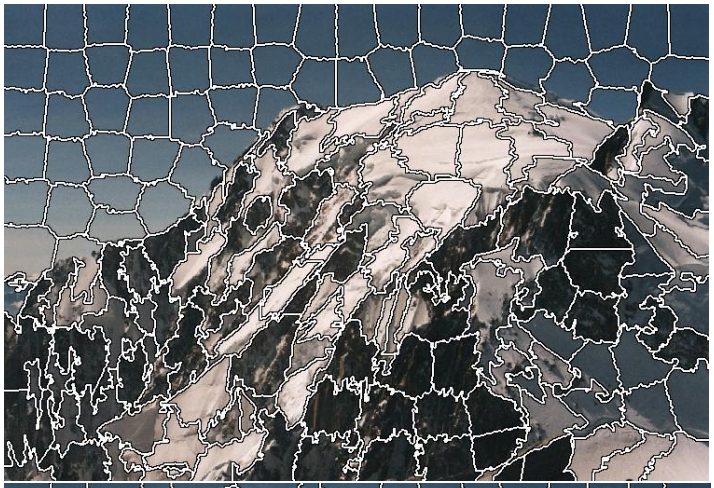
Number of superpixels are choosen as 200. Compactness factor is choosen as 20. I believe that these numbers are best for all images because some images are plain and have little color transitions and have few in the corners Therefore high number of superpixelation would be unnecessary. Also compactness factor were choosen as 20 because, although some images are plain there were some places that edges and transitions were complex, which means that circle segmentation would not reflect the right context of image.

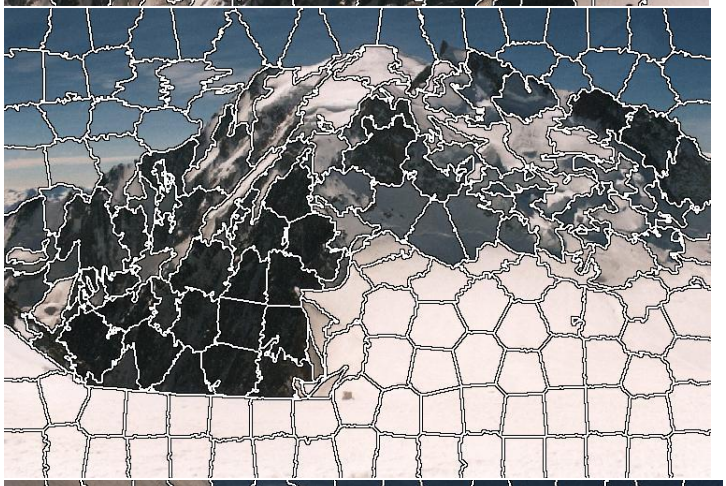
2) Segmentation results (SLIC is used)

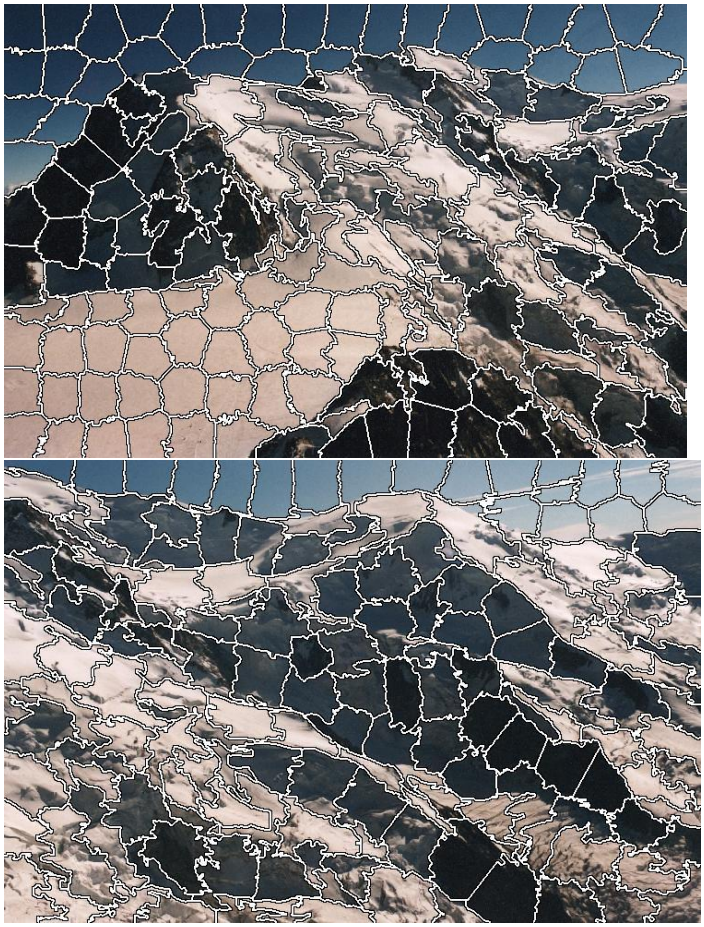










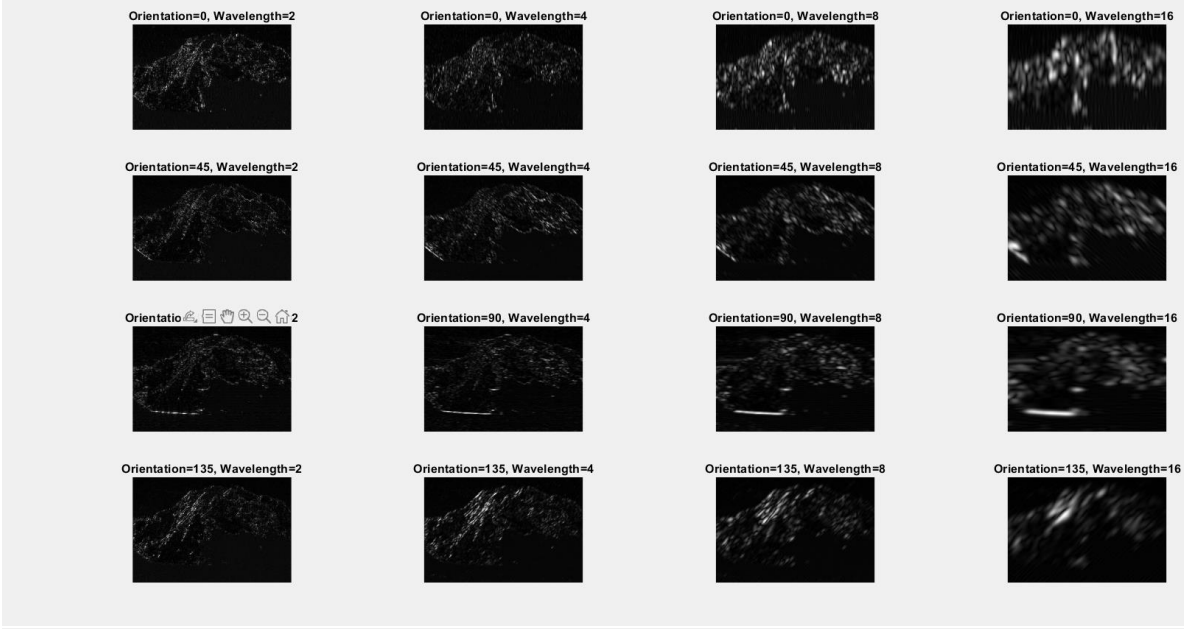


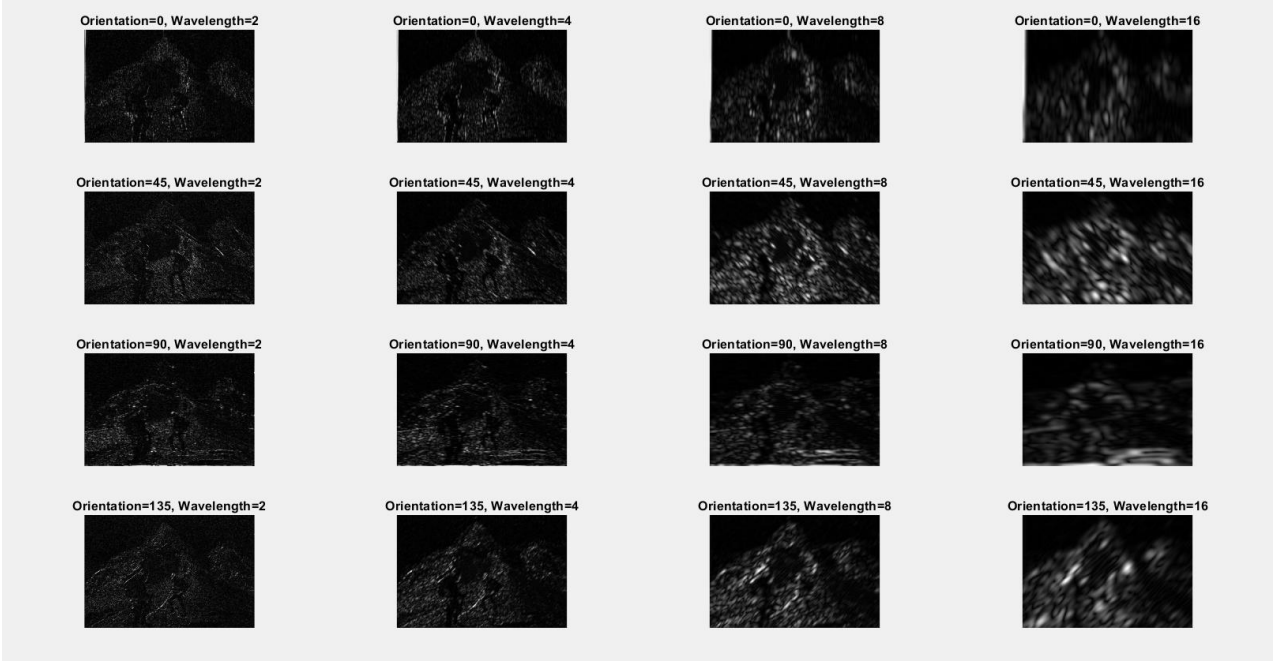
3) Description of the parameters used for Gabor texture feature extraction.

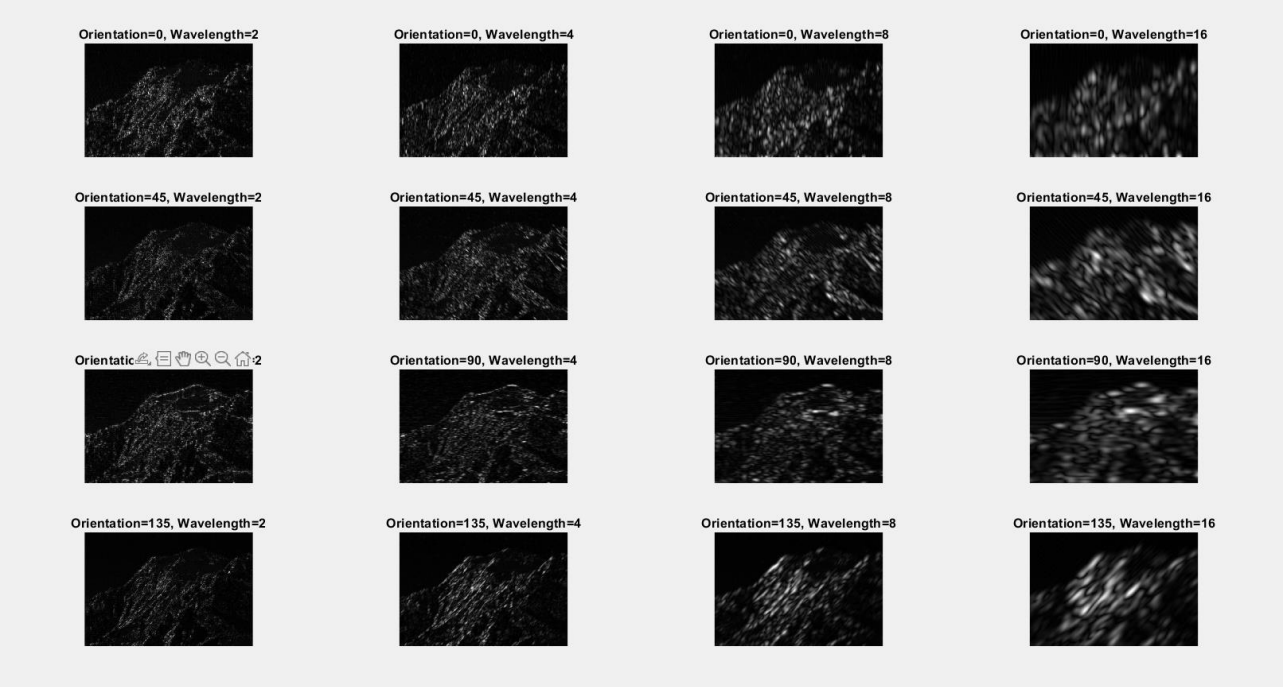
Wavelength is used as : 2,4,8,16 respectively
Orientation is used as: 0,45,90,135

As it can be seen in part 4 of this report. Lower wavelength produces more detailed and finer images on the other hand while wavelength increases such as 8 or 16. Detailes of the images becomes hard to identifiy. In addition to this, orientation values tells the filter to which direction that the filter should be sensitive. According to my results the value of orientation , 135 fits best compared to 0 , 45 , 90 values in many images.

4) Gabor texture feature examples


Orientation=0, Wavelength=2
Orientation=0, Wavelength=4
Orientation=0, Wavelength=8
Orientation=0, Wavelength=16
Orientation=45, Wavelength=2
Orientation=45, Wavelength=4
Orientation=45, Wavelength=8
Orientation=45, Wavelength=16
Orientatio 2
Orientation=90, Wavelength=4
Orientation=90, Wavelength=8
Orientation=90, Wavelength=16
Orientation=135, Wavelength=2
Orientation=135, Wavelength=4
Orientation=135, Wavelength=8
Orientation=135, Wavelength=16


Orientation=0, Wavelength=2
Orientation=0, Wavelength=4
Orientation=0, Wavelength=8
Orientation=0, Wavelength=16
Orientation=45, Wavelength=2
Orientation=45, Wavelength=4
Orientation=45, Wavelength=8
Orientation=45, Wavelength=16
Orientation=90, Wavelength=2
Orientation=90, Wavelength=4
Orientation=90, Wavelength=8
Orientation=90, Wavelength=16
Orientation=135, Wavelength=2
Orientation=135, Wavelength=4
Orientation=135, Wavelength=8
Orientation=135, Wavelength=16



5) Any source code that I wrote for the assignment.

```
imageDir = 'C:\Users\user\Desktop\Cs484_Hw3';
superpixelDir = 'C:\Users\user\Desktop\datfiles';
gaborSaveDir = 'C:\Users\user\Desktop\gabor';
wavelength = 4;
orientation = 90;
gaborArray = gabor([2 4 8 16],[0 45 90 135]);
all_features = []; % Store all feature vectors
% Loop through all images.
for i = 1:10
image_path = fullfile(imageDir, [num2str(i) '.jpg']);
superpixel_path = fullfile(superpixelDir, [num2str(i) '.dat']);
gray_img = processImage(image_path);
[mag, phase] = getGaborFilters(gray_img, wavelength, orientation);
gaborMag = imgaborfilt(gray_img,gaborArray);
saveGaborMagnitude(gaborMag, gaborSaveDir, i);
fileID = fopen(superpixel_path, 'r'); % open the file
superpixels = fread(fileID, [10, 50], 'int'); % replace num_rows and num_cols with the actual size
fclose(fileID); % close the file
```

```
feature_matrix = getSuperpixelFeatures(superpixels, gaborMag);
all_features = [all_features; feature_matrix]; % append feature vectors to all_features
% Now you have a feature_matrix for each image, do something with it...
% For example, save it to a file:
save(fullfile(gaborSaveDir, ['feature_matrix_', num2str(i), '.mat']), 'feature_matrix');
showImages(gray_img, mag, phase);
showGaborFilters(gaborMag, gaborArray);
end
% Now that you've processed all images and accumulated all feature vectors,
% you can perform clustering on all_features.
% number of clusters
k = 5; % you should choose a suitable number
% perform k-means clustering
[cluster_idx, cluster_centroid] = kmeans(all_features, k);
% ... the rest of your code ...
function img = processImage(image_path)
img = imread(image_path);
img = im2gray(img);
end
function [mag, phase] = getGaborFilters(img, wavelength, orientation)
[mag, phase] = imgaborfilt(img, wavelength, orientation);
end
function saveGaborMagnitude(gaborMag, saveDir, index)
save(fullfile(saveDir, ['gaborMag_', num2str(index), '.mat']), 'gaborMag');
end
function feature_matrix = getSuperpixelFeatures(superpixels, gaborMag)
num_superpixels = max(superpixels(:));
num_filters = size(gaborMag, 3);
% Initialize feature matrix
feature_matrix = zeros(num_superpixels, num_filters);
% For each superpixel, average Gabor features of all pixels within it
for i = 1:num_superpixels
for j = 1:num_filters
```

```
feature_matrix(i, j) = mean(gaborMag(superpixels == i, j));
end
end
end
function showImages(img, mag, phase)
figure
tiledlayout(1,3)
nexttile
imshow(img)
title('Original Image')
nexttile
imshow(mag,[])
title('Gabor Magnitude')
nexttile
imshow(phase,[])
title('Gabor Phase')
end
function showGaborFilters(gaborMag, gaborArray)
figure
for p = 1:16
subplot(4,4,p)
imshow(gaborMag(:,:,p),[]);
theta = gaborArray(p).Orientation;
lambda = gaborArray(p).Wavelength;
title(sprintf('Orientation=%d, Wavelength=%d',theta,lambda));
end
End
```

6) Citation for any external code used.

[1] “Wavelength,” Create Gabor filter or Gabor filter bank - MATLAB, https://www.mathworks.com/help/images/ref/gabor.html (accessed May 24, 2023).