DLCV Homework #1

r08944022 蔡仲閔

vtsai01@cmlab.csie.ntu.edu.tw

().

Problem 1: K-Means Clustering

1. ***For K = 2, 4, 8, 16, and 32, perform K-means clustering on the provided bird.jpg by taking the RGB values of each pixel as the feature of interest.***

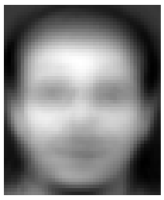
2. ***Repeat 1. but take both RGB values and the location (x and y) as a five dimensional vectoras the feature for describing each pixel. Show the segmentation results.***
   To normalize five dimension, we scale x and y to $[0, 255]$

3. ***Compare your results obtained in 1. and 2., and briefly explain the differences between the two methods under the same K. If further improved segmentation results would be desirable, please provide possible modification or extension to the above feature definition (and visualize your results).***
   Compare the two result above. We could see that the segmentation do not perform well when add spatial features(x and y). Result 2 take more effort in the background instead of the head of bird which we care more about.
   To improve the result, we convert RGB color channel into ***Lab*** (https://en.wikipedia.org/wiki/CIELAB_color_space) color space and then convert *Cartesian coordinate system (x, y)* into ***Polar coordinate system***.

Note: we didn't do normalize since the location should not have same weight as color space.

Problem 2: Principal Component Analysis

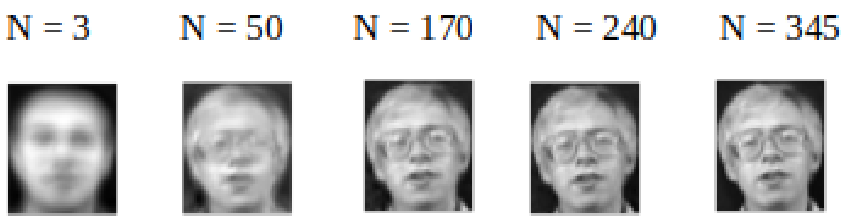
1. ***Perform PCA on the training set. Plot the mean face and the first four eigenfaces.***



Mean Face

First 4 eigenfaces

2. ***Take $person_2image_1$, and project it onto the PCA eigenspace you obtained above. Reconstruct this image using the first n = 3, 50, 170, 240, 345 eigenfaces. Plot the five reconstructed images.***



Reconstruct this image using the first n = 3, 50, 170, 240, 345 eigenfaces

3. ***For each of the five images you obtained in 2., compute the mean squared error (MSE) between the reconstructed image and the original image. Record the corresponding MSE values in your report.***

n eigenfaces	**3**	**50**	**170**	**240**	**345**
MSE	0.0114	0.0036	0.0007	0.0002	3.31e-06

4. ***Now, apply the k-nearest neighbors algorithm to classify the testing set images. First,you will need to determine the best k and n values by 3-fold cross-validation. For simplicity, the choices for such hyperparameters are k = {1, 3, 5} and n = {3, 50, 170}. Show the cross-validation results and explain your choice for (k, n).***
   The value in the following table are the average accuracy from cross-validation results. From the table, we will pick n=50 and k=1 for the next part.

k nearest neighbor/n eigenfaces	3	50	170
1	0.67	0.90	0.54
3	0.57	0.81	0.25
5	0.51	0.78	0.20

5. ***Use your hyperparameter choice in 4. and report the recognition rate of the testing set.***
   Accuracy: 0.875

Problem 3: Image Filter

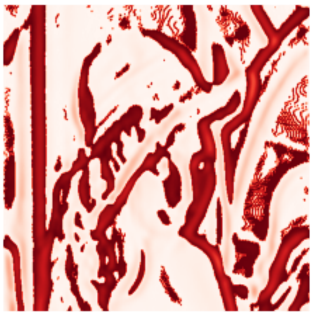
1. ***Implement a discrete 2D Gaussian filter using a 3 * 3 kernel with $\sigma = \frac{1}{2ln2}$. Use the provided lena.png as input, and plot the output image in your report. Briefly describe the effect of the filter.***
   After apply Gaussian filter on the image, the image become more smooth. In other word, the image lost some details in high frequency. This effect is similar to low-pass filter when we talk about signal processing.

Image after using Gaussian filter

2. ***Write down your answers of k_x and k_y . Also, plot the resulting images I_x and I_y using the provided `lena.png` as input.***

i =	x	y
k_i	$k_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$	$k_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$
I_i		

3. ***Plot the two output gradient magnitude images in your report. Briefly explain the differences in the results.***

input image	**Oringinal**	**Gaussian($\sigma = \frac{1}{2ln2}$)**	**Gaussian($\sigma = 5$)**
I_i			

From the table above, we can see that after applying gaussian filter, only the main edges left. Those high frequency noise are no longer remains. These effects can easily observe when the σ of gaussian filter become larger.

tags: `NTU` `Homework` `DLCV`