Deep Learning for Computer Vision HW2 Report

電機三 張景程 b04901138

Problem 1 Kernel Trick

kernel function $K(x, x') = (x^T x)^2$

Given $x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$, $x' = \begin{bmatrix} x_1' \\ x_2' \end{bmatrix}$.

$$K(x, x') = (x_1 x_1' + x_2 x_2')^2 = (x_1 x_1')^2 + 2(x_1 x_1')(x_2 x_2') + (x_2 x_2')^2 .$$

$$(x_1 x_1')^2 + 2(x_1 x_1')(x_2 x_2') + (x_2 x_2')^2 = \begin{bmatrix} x_1^2 & \sqrt{2} x_1 x_2 & x_2^2 \end{bmatrix} \begin{bmatrix} x_1'^2 \\ 2 x_1' x_2' \\ x_2'^2 \end{bmatrix}$$

$$= \Phi(x)^T \Phi(x')$$

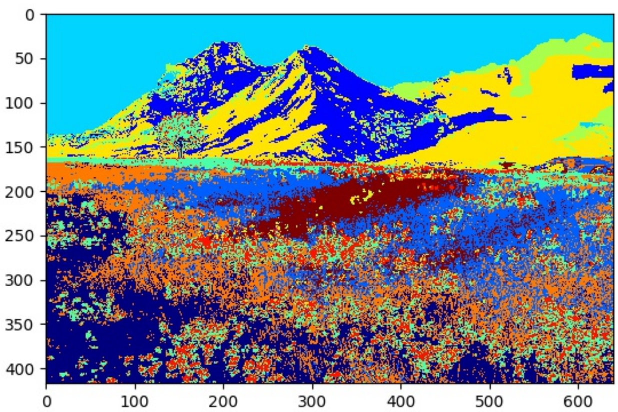
$$\Rightarrow \Phi(x) = \Phi\left(\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}\right) = \begin{bmatrix} x_1^2 \\ \sqrt{2} x_1 x_2 \\ x_2^2 \end{bmatrix} \#.$$

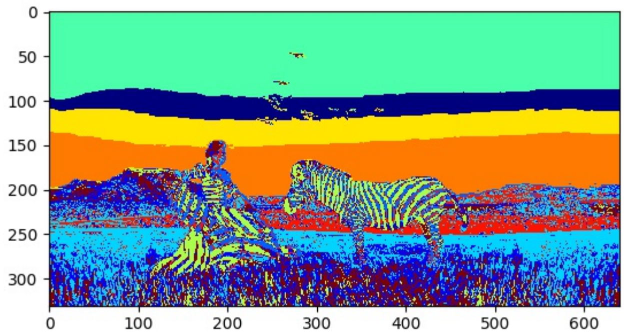
Problem 2 Color and Texture Segmentation

Original Picture :

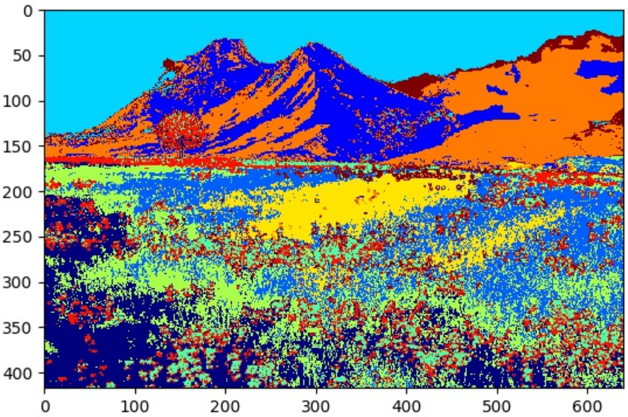
(a) Color Segmentation

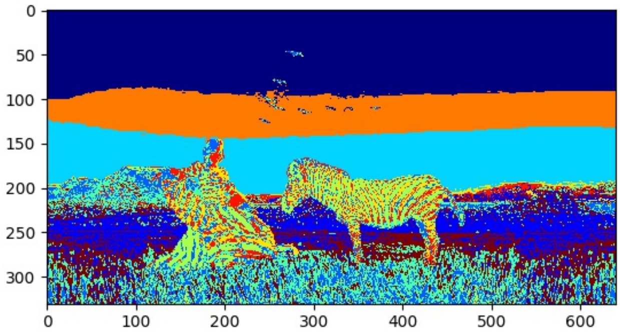
RGB :





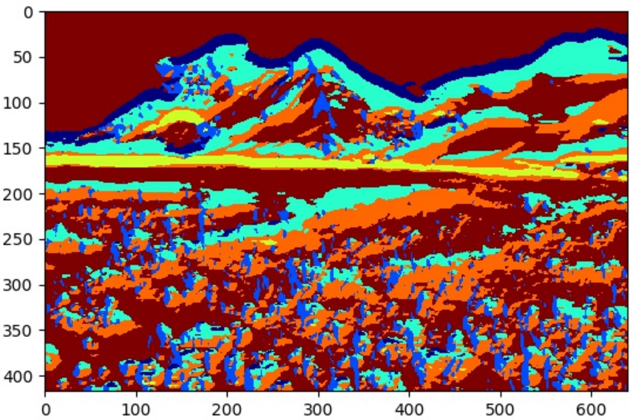
LAB :

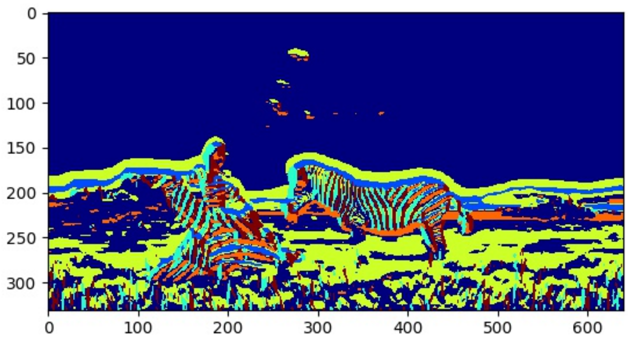




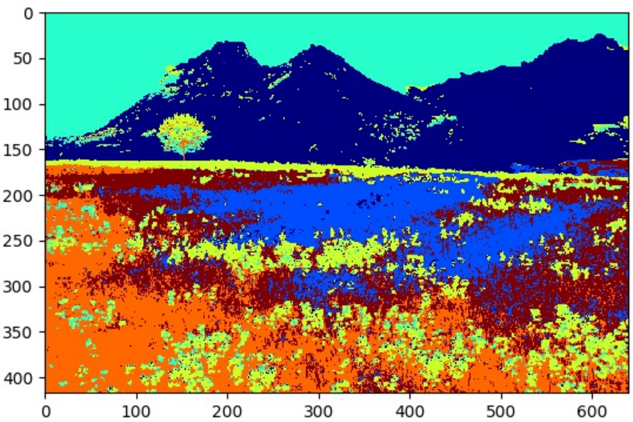
(b) Texture Segmentation

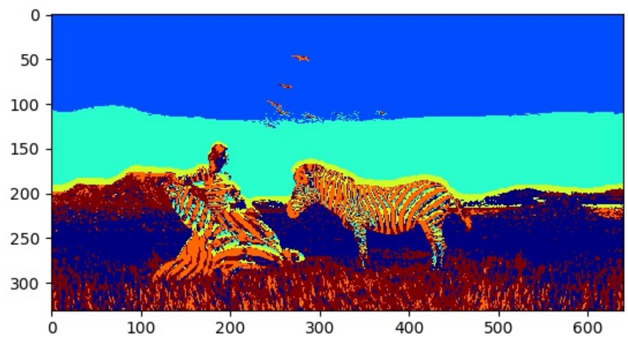
38 dimensions :





41 dimensions (+LAB) :



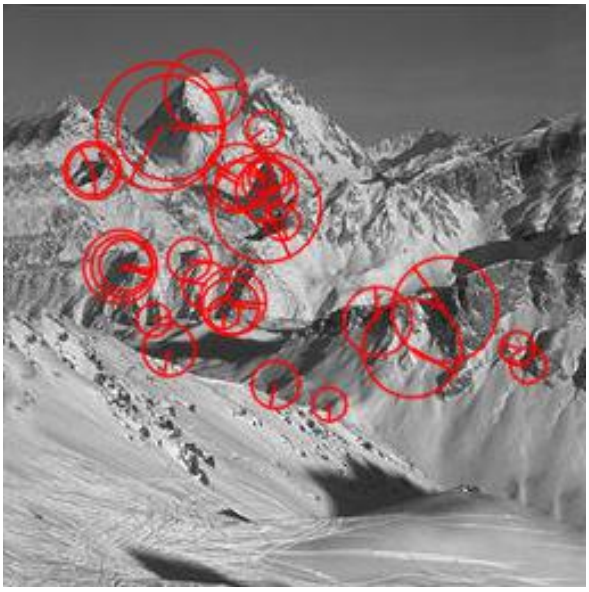


Problem 3 Recognition with Bag of Visual Words

(a) Plot the interest points detection results

Image picked : *Mountain/image_0033.jpg*

Indicate 30 interest points

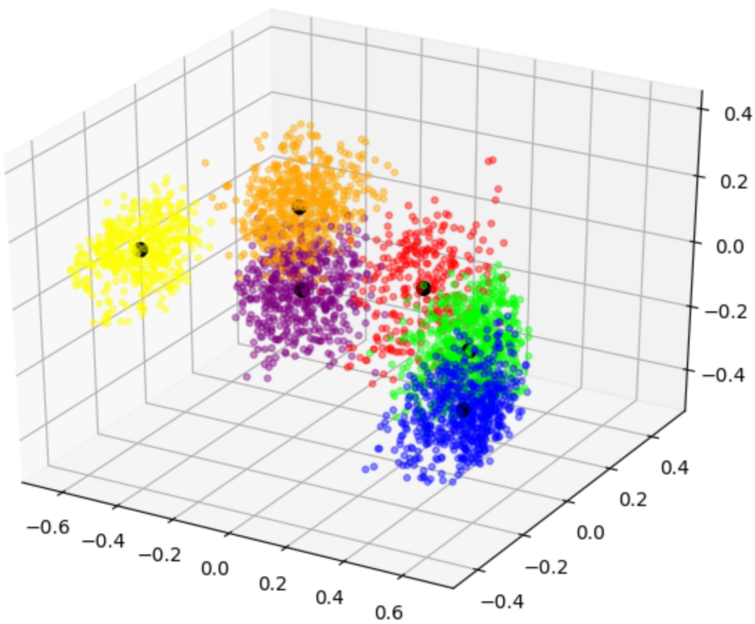


(b) Plot the visual words and the associated interest points in the PCA supspace

C clusters : 50

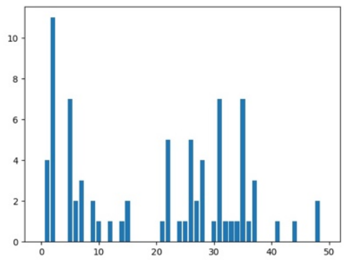
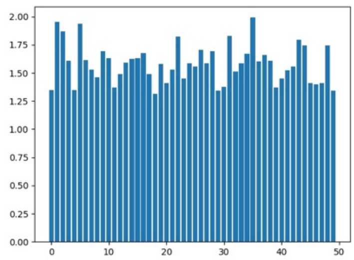
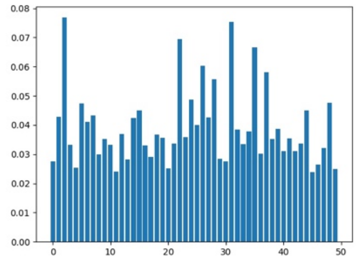
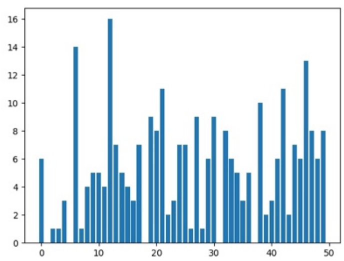
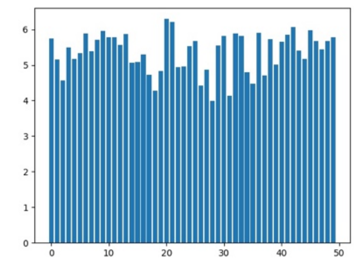
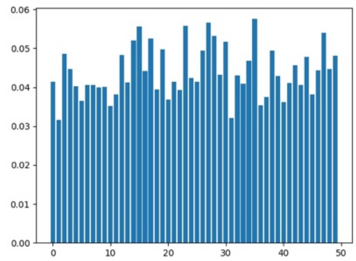
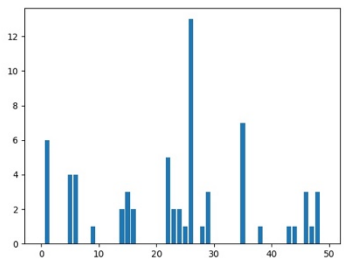
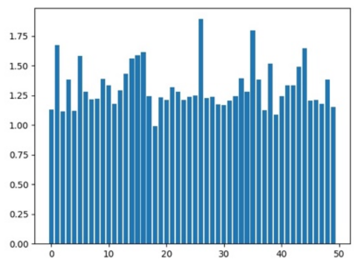
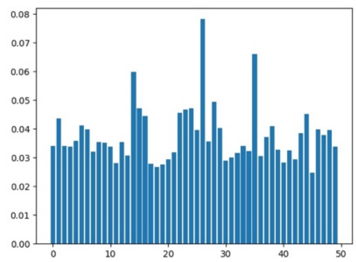
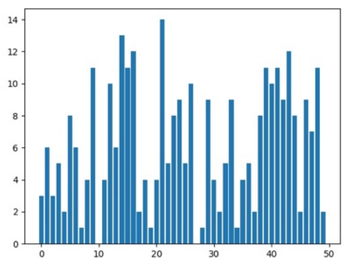
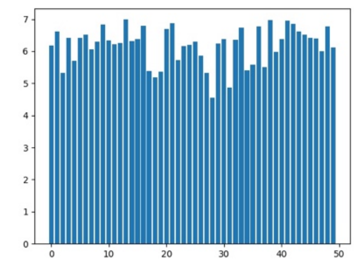
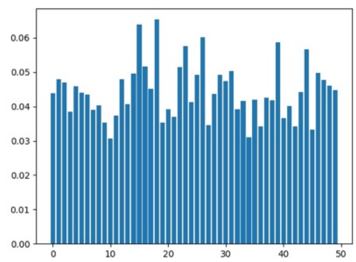
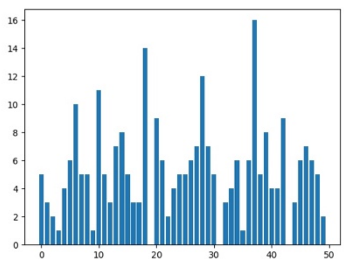
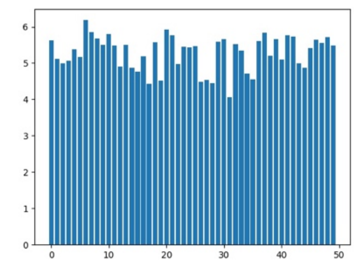
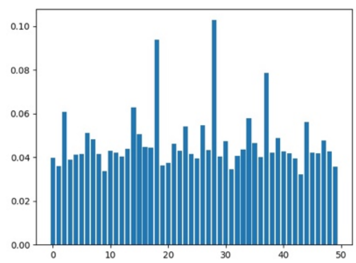
Max Iterations = 5000

In each cluster, there is a black point which represents the corresponding centroid.



(c) Choose one image from each category and plot their Hard-Sum, Soft-Sum, Soft-Max results.

C clusters : 50 Max Iteration = 5000 Hessian Threshold = 1000

	Hard-Sum	Soft-Sum	Soft-Max
Coast Image_ 0022.jpg			
Forest Image_ 0020.jpg			
Highway Image_ 0018.jpg			
Mountain Image_ 0044.jpg			
Suburb Image_ 0016.jpg			

From above histograms, we can observe that the distribution of Hard-Sum strategy has several peaks on some specific visual words and has significant difference between different classes. On the other hand, the difference between classes of the distribution in Soft-Sum and Soft-Max is not as big as the Hard-Sum one. It means that these two strategy may not get the strong characteristic of pictures. Therefore, I would expect that the **Hard-Sum** strategy will have a better classification result.

(d) Adopt the k-nearest neighbors classifier (k-NN) to perform classification using the above Bow features.

(1) Use Train-10 for training and Test-100 for testing. (select k=5)

C clusters : 50 Max Iteration = 5000 Hessian Threshold = 1000

	Hard-Sum	Soft-Sum	Soft-Max
Accuracy	0.578	0.356	0.584

由結果可以看出 Hard-Sum 和 Soft-Max 的正確率非常接近，而 Soft-Sum 準確率最差，推論其原因應為 Hard-Sum 只保留了幾個最重要的特徵(characteristic)，使得有幾項數據皆為零，因此可能會把其他的資訊也一併濾掉；而從 Soft-Max 的 histogram 可以看出，它除了仍有少數幾個 peak 以外，其他幾項數值皆不為零，也就是可以把一些 background 的資訊保留下來，因此 Soft-Max 表現也很好。

Soft-Sum 的 distribution 結果會使每個 class 之間的差距相差不大，因此機器也較難正確的分辨類別。

(2) Use Train-100 for training and Test-100 for testing.

C clusters : 50 Max Iteration = 5000 Hessian Threshold = 1000

k / strategy	Hard-Sum	Soft-Sum	Soft-Max
1-nearest	0.660	0.598	0.588
3-nearest	0.686	0.620	0.648
5-nearest	0.698	0.634	0.674
7-nearest	0.714	0.634	0.676

C clusters : 50 Max Iteration = 10000 Hessian Threshold = 1000

k / strategy	Hard-Sum	Soft-Sum	Soft-Max
1-nearest	0.644	0.596	0.614
3-nearest	0.668	0.620	0.646
5-nearest	0.686	0.632	0.678
7-nearest	0.706	0.634	0.670

C clusters : 50　　Max Iteration = 5000　　Hessian Threshold = 500

k / strategy	Hard-Sum	Soft-Sum	Soft-Max
1-nearest	0.686	0.618	0.646
3-nearest	0.726	0.620	0.666
5-nearest	0.744	0.630	0.694
7-nearest	0.728	0.642	0.704

C clusters : 100　　Max Iteration = 5000　　Hessian Threshold = 1000

k / strategy	Hard-Sum	Soft-Sum	Soft-Max
1-nearest	0.646	0.600	0.652
3-nearest	0.644	0.616	0.676
5-nearest	0.666	0.642	0.704
7-nearest	0.674	0.636	0.694

C clusters : 100　　Max Iteration = 10000　　Hessian Threshold = 1000

k / strategy	Hard-Sum	Soft-Sum	Soft-Max
1-nearest	0.652	0.602	0.638
3-nearest	0.658	0.610	0.652
5-nearest	0.650	0.640	0.670
7-nearest	0.672	0.634	0.696

C clusters : 100　　Max Iteration = 5000　　Hessian Threshold = 500

k / strategy	Hard-Sum	Soft-Sum	Soft-Max
1-nearest	0.704	0.624	0.670
3-nearest	0.718	0.624	0.708
5-nearest	0.712	0.632	0.718
7-nearest	0.714	0.638	0.708

從結果可以看出當我們把 training data 從 train-10 改為 train-100 之後，三種 BoW 的準確率都提升不少，因為我們 training data 數量增為十倍，可以從資料中得到更多的資訊，也因此會得到較高的準確率。

此外，由實驗數據中也可以觀察出，增加 cluster 的數量 C，Hard-Sum 的準確率會降低而 Soft-Max 的準確率則會增加；k 的值越大(5 or 7)，對三種 BoW 的準確率也都會有所提升；而增加 kmeans 的 iteration 數量會使準確率略微增加，但效果並不明顯，

理論上 iteration 次數越多，所找到的 centroid 會更準確，可能是因為 Iteration 5000 次之後已經收斂差不多了，所以和一萬次差別不大，若是數量改為 1000 v.s. 5000 可能會得到更明顯的結果。

而 Hessian Threshold 設得越小，會得到更多的 descriptor，所以在 detect 一張 image 時可以得到更多資訊，也因此準確率會越高，但也因為 descriptor 數量增加，載運算上也會花更多時間。