Deep Learning for Computer Vision

Homework2

R06942052 電信碩一 鍾勝隆

Problem 1: Kernel Trick

A:

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

Let $x = [x_1, x_2]^T$ and $x' = [x'_1, x'_2]^T$

$$(x^T x')^2 = \left(\begin{bmatrix} x_1 & x_2 \end{bmatrix} \begin{bmatrix} x'_1 \\ x'_2 \end{bmatrix} \right)^2$$

$$= (x_1 x'_1 + x_2 x'_2)^2$$
$$= (x_1 x'_1)^2 + 2 x_1 x'_1 x_2 x'_2 + (x_2 x'_2)^2$$

$$= x_1^2 x'^2_1 + 2 x_1 x_2 x'_1 x'_1 + x_2^2 x'^2_2$$

$$= \begin{bmatrix} x_1^2 & \sqrt{2 x_1 x_2} & x_2^2 \end{bmatrix} \begin{bmatrix} x'^2_1 & \sqrt{2 x'_1 x'_2} & x'^2_2 \end{bmatrix}^T$$
$$= \Phi(x)^T \Phi(x')$$

Because $\Phi(x) \in \mathbb{R}^3$,

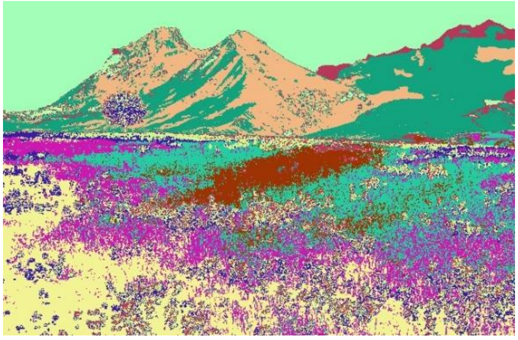
$$\Phi(x) = \begin{bmatrix} x_1^2 & \sqrt{2 x_1 x_2} & x_2^2 \end{bmatrix}^T$$

Problem 2: Color and Texture Segmentation

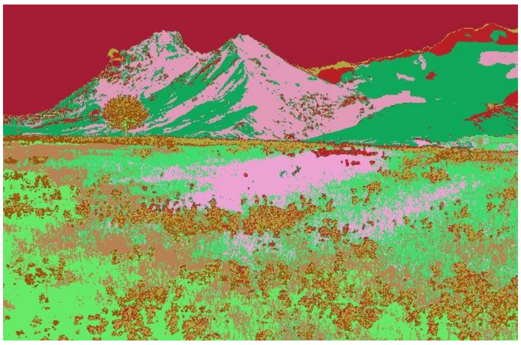
A:

a. Color segmentation:

RGB Color Space:

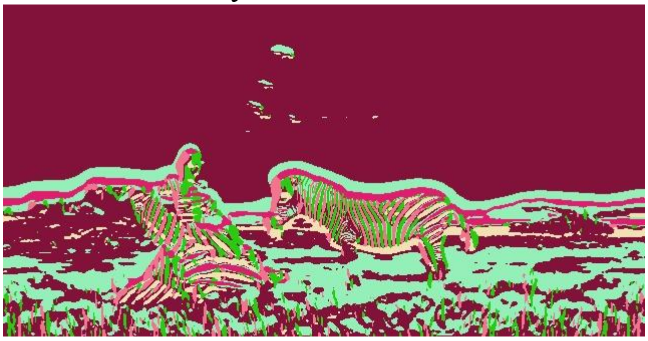



Lab Color Space:

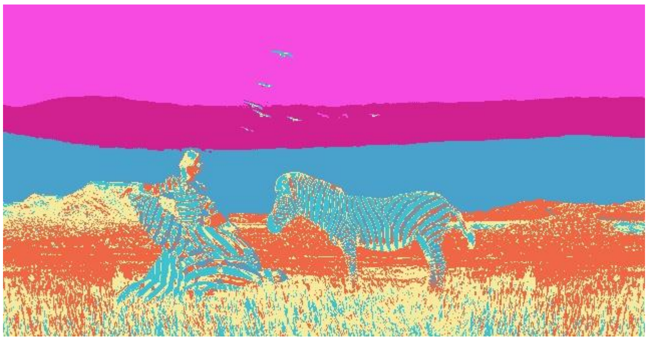
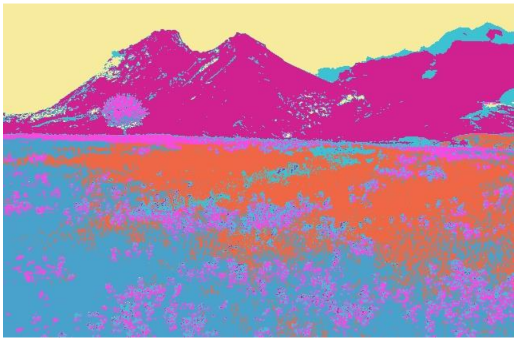



b. Texture segmentation:

Texture Only



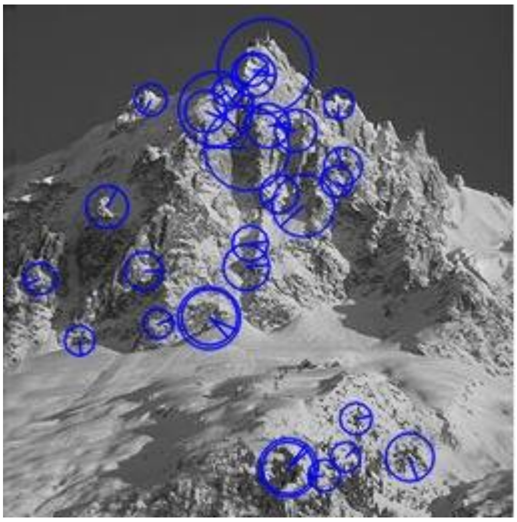

Textrue + Color

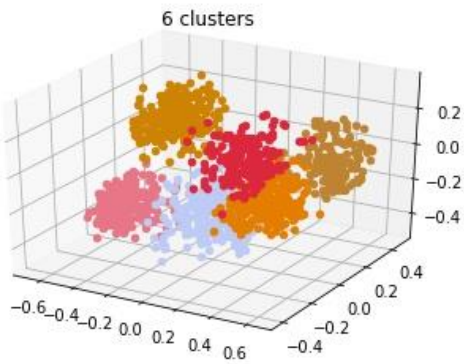
Problem 3: Recognition with Bag of Visual Words

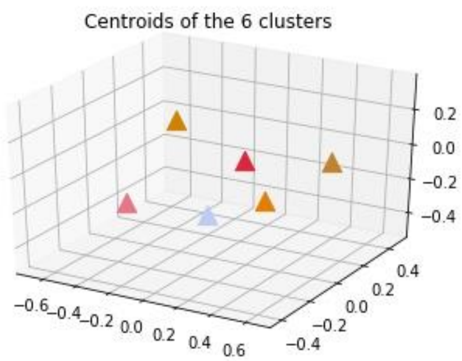
A:

a. Randomly pick an image from **Train-10**. Plot the interest point detection result with SURF. The total number of detected points is 30. The size of the circles and radius highlighted is the descriptor of SURF.

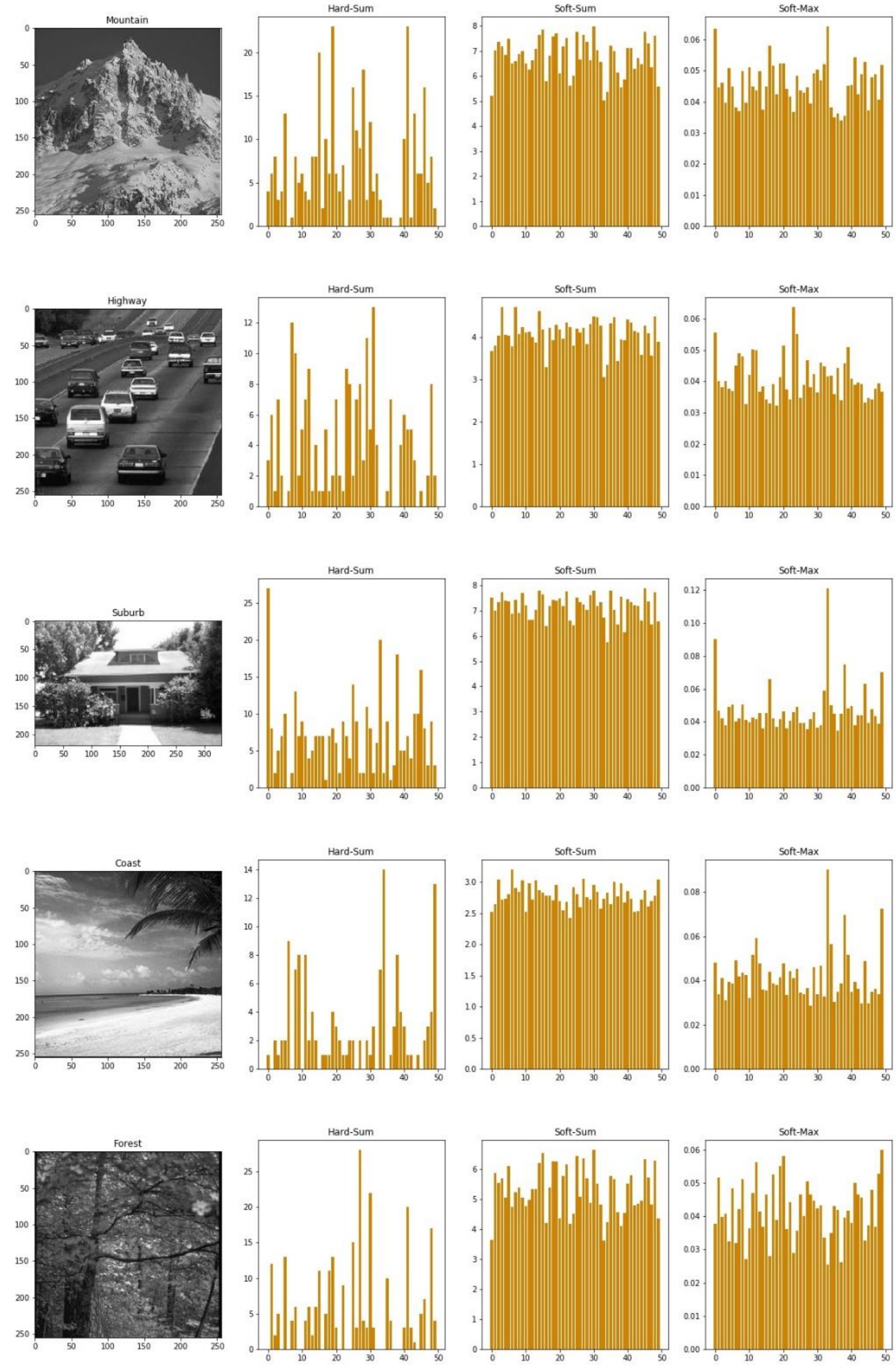


b. 3D PCA projection of the 6 visual words out of 50 words
The triangle markers are the centroids of the kmeans algorithm which represent the visual words. The average number of the key points in each image in Train-10 is 400.32.





c. Choose one image from each category and plot their **Hard-Sum**, **Soft-Sum**, and **Soft-Max**, respectively.
From the following result, I think it is more distinguishable with the Hard-Sums because I can tell the difference in the Bag of word space. I expect the **Hard-Sum** will be the better strategy.


Mountain
Hard-Sum
Soft-Sum
Soft-Max
Highway
Hard-Sum
Soft-Sum
Soft-Max
Suburb
Hard-Sum
Soft-Sum
Soft-Max
Coast
Hard-Sum
Soft-Sum
Soft-Max
Forest
Hard-Sum
Soft-Sum
Soft-Max

d. (i)Use **Train-10** as the training data and **Test-100** for testing. Report the classification accuracy using Hard-Sum, Soft-Sum, and Soft-Max.

C = 50

K	Hard-Sum	Soft-Sum	Soft-Max
1	**53.6%**	39.8%	52.6%
2	42.0%	32.0%	**47.4%**
3	**53.2%**	32.6%	51.6%
4	**54.4%**	31.2%	49.6%
5	**56.4%**	32.0%	50.0%
6	**55.6%**	30.6%	47.2%
7	**57.2%**	33.4%	48.6%
8	**58.0%**	32.6%	47.0%
9	**56.2%**	30.2%	49.2%
10	**54.4%**	38.4%	46.8%

With most of the K, **Hard-Sum** strategy performs better than the other strategies. However, **Soft-Max** is not far behind it. I believe **Soft-Max** bag of word can precisely describe the images better than **Hard-Sum** if we tuning more parameter.

(ii) Repeat the previous steps using **Train-100** as the training data

C = 20

K	Hard-Sum	Soft-Sum	Soft-Max
1	**64.6%**	57.2%	49.8%
3	**66.6%**	60.0%	52.4%
5	**67.8%**	58.2%	54.6%
7	**70.8%**	58.2%	57.6%
9	**71.0%**	59.6%	57.8%

C = 50

K	Hard-Sum	Soft-Sum	Soft-Max
1	**65.8%**	57.0%	57.6%
3	**70.0%**	60.2%	57.6%
5	**69.6%**	60.6%	59.2%
7	**71.2%**	59.8%	59.6%
9	**69.0%**	58.8%	60.4%

C = 100

K	Hard-Sum	Soft-Sum	Soft-Max
1	**65.6%**	58.4%	60.8%
3	**67.2%**	60.8%	64.0%
5	**70.2%**	60.0%	65.6%
7	**68.4%**	58.8%	63.8%
9	**67.8%**	58.2%	64.8%

C= 200

K	Hard-Sum	Soft-Sum	Soft-Max
1	59.4%	57.6%	**64.4%**
3	59.0%	60.4%	**67.6%**
5	60.0%	59.4%	**64.8%**
7	60.4%	59.4%	**66.4%**
9	60.8%	58.2%	**65.4%**

I use C = [20, 50, 100, 200] and K = [1, 3, 5, 7, 9] as combination to do the experiment with **Train-100** dataset. From the accuracy shown above, I found it is helpful for classification to use more data to do the training. (50->500). The best accuracy I had achieve is **71.2%** using the 50 bag-of-words with Hard-Sum strategy. It is worth mentioning that when C = 200, **Soft-Max** is better than Hard-Sum. Soft-Max seems to be stable if the another parameter is badly chosen.