EE569: Homework #6

Issued: 04/14/2022
Due: 11:59PM, 05/01/2022

General Instructions:

1. Read *Homework Guidelines* for the information about homework programming, write-up and submission. If you make any assumptions about a problem, please clearly state them in your report.
2. You need to understand the USC policy on academic integrity and penalties for cheating and plagiarism. These rules will be strictly enforced.
3. You are required to use *Python* for this Homework.
4. This content is protected and may not be shared, uploaded, or distributed.

Problem 1: Origin of Green Learning (GL) (35%)

(a) Feedforward-designed Convolutional Neural Networks (FF-CNNs) (20%)

When two CNN layers are cascaded, a non-linear activation function is used in between. As an alternative to the non-linear activation, Kuo *et al.* proposed the Saak (subspace approximation via augmented kernels) transform [1] and the Saab (subspace approximation via adjusted bias) transform [2]. Specifically, Kuo *et al.* [2] proposed the first example of a feedforward-designed CNN (FF-CNN), where all model parameters are determined in a feedforward manner without backpropagation. It has two main cascaded modules:

1) Convolutional layers via multi-stage Saab transforms

2) Fully-connected (FC) layers via multi-stage linear least squared regression (LLSR)

Although the term "successive subspace learning" (SSL) was not used in [2] explicitly, it does provide the first SSL design example.

Read paper [2] carefully and answer the following questions:

  (1) Summarize the Saab transform with a flow diagram. Explain it in your own words. The codes for the Saab transform can be found at https://github.com/USC-MCL/Channelwise-Saab-Transform. Please read the codes with the paper to understand the Saab transform better.

  (2) Explain similarities and differences between FF-CNN and backpropagation-designed CNN (BP-CNNs).

Do not copy any sentence from a paper directly which is plagiarism. Your scores will depend on the degree of your understanding.

(b) Understanding PixelHop and PixelHop++ (15%)

Two interpretable models adopting the SSL principle for the image classification task were proposed by Chen *et al.* They are known as PixelHop [3] and PixelHop++ [4]. Read the two papers carefully and answer the questions below. You can use various tools in your explanation such as flow charts, figures, formula etc. You should demonstrate your understanding through your answer.

  (1) Explain the SSL methodology in your own words. Compare Deep Learning (DL) and SSL.

  (2) What are the functions of Modules 1, 2 and 3, respectively, in the SSL framework?

(3) Explain the neighborhood construction and subspace approximation steps in the PixelHop unit and the PixelHop++ unit and make a comparison. Specifically, explain the differences between the basic Saab transform and the channel-wise (c/w) Saab transform.

Problem 2: PixelHop & PixelHop++ for Image Classification (65%)

Please apply PixelHop and PixelHop++ to the MNIST and the Fashion-MNIST datasets.

(a) Building PixelHop++ Model (35%)

The block diagram of PixelHop++ is shown in Figure 1. It contains three PixelHop++ Units. The codes for the c/w Saab transform module is provided in the GitHub. You can import them in your program to build your model in Python based on the diagram. You should adopt the parameters and the classifier choice in Table 1.
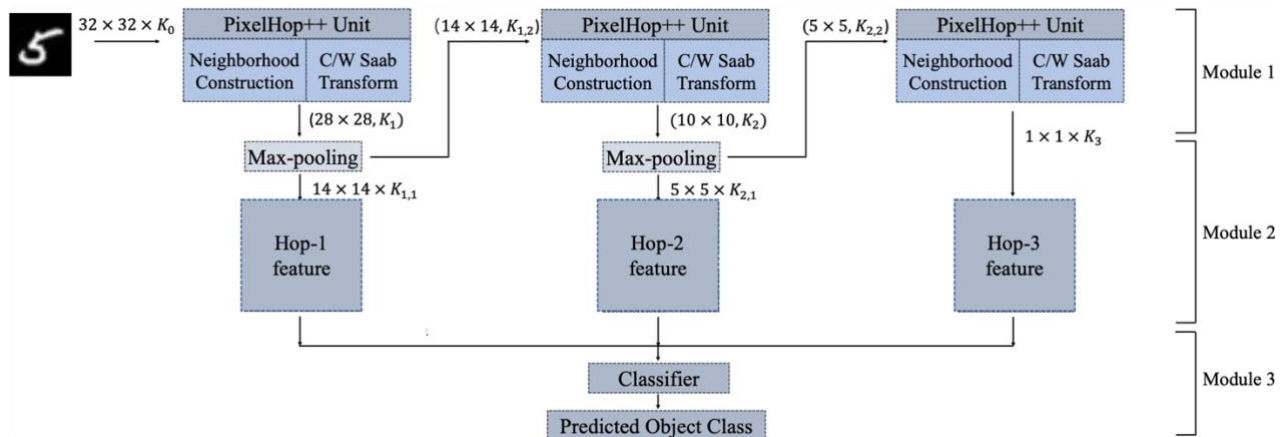

Figure 1 Block diagram of the PixelHop++ model [4]

Table 1 Choice of hyper-parameters of PixelHop++ model for MNIST dataset

Spatial Neighborhood size in all PixelHop++ units	5x5
Stride	1
Max-pooling	(2x2) -to- (1x1)
Energy threshold for intermediate nodes (*TH1*)	0.005
Energy threshold for discarded nodes (*TH2*)	0.001
Classifier	XGBoost
Number of estimators in classifier	100

(1) Train Module 1 using the whole set or a subset of 10000 training images (depending on your memory). Remember to keep balance among different classes (i.e. randomly select 1000 images per class if you use 10,000 training images). Then, train Module 3 **only on Hop3 feature**. Report training time and train accuracy. What is your model size in terms of the total number of parameters?

(2) Apply your model to 10,000 testing images and report the test accuracy.

(3) With the same TH2, try different TH1 energy threshold values in Module 1 and report the test

accuracy and the model size for different choices. Plot the curve of TH1 vs. the test accuracy. Discuss your result.

(b) Comparison between PixelHop and PixelHop++ (15%)

The codes for the Saab transform are provided in the GitHub. Please use the Saab transform (instead of c/w Saab transform) to build the PixelHop model with the same parameter settings as PixelHop++ in Table 1. Note that TH2 is treated as the energy threshold used in the PixelHop paper.

(1) Compare the performance of PixelHop and PixelHop++ in terms of the train accuracy and the test accuracy. Discuss your result.

(2) Compare the model size of PixelHop and PixelHop++ in terms of the number of model parameters. Discuss your result.

(c) Error analysis (15%)

A dataset often contains easy and hard classes. Conduct the following error analysis based on your trained PixelHop++ model using 60,000 training images:

(1) Compute the confusion matrix and show it as a heat map in your report. Which object class yields the lowest error rate? Which object class yields the highest one?

(2) Find out the confusing class groups and discuss why they are easily confused with each other. You can use some exemplary images to support your statement.

(3) Propose ideas to improve the accuracy of difficult classes for PixelHop++ and justify your ideas. There is no need to implement your ideas.

References

[1] C.-C. Jay Kuo and Yueru Chen, "On data-driven Saak transform," *Journal of Visual Communication and Image Representation*, vol. 50, pp. 237–246, 2018.

[2] C.-C. Jay Kuo, Min Zhang, Siyang Li, Jiali Duan, and Yueru Chen, "Interpretable convolutional neural networks via feedforward design," *Journal of Visual Communication and Image Representation*, vol. 60, pp. 346–359, 2019.

[3] Yueru Chen and C.-C. Jay Kuo, "Pixelhop: A successive subspace learning (ssl) method for object recognition," *Journal of Visual Communication and Image Representation*, p. 102749, 2020.

[4] Yueru Chen, Mozhdeh Rouhsedaghat, Suya You, Raghuveer Rao, C.-C. Jay Kuo, "PixelHop++: A Small Successive-Subspace-Learning-Based (SSL-based) Model for Image Classification," *https://arxiv.org/abs/2002.03141*, 2020

[5] Yueru Chen, Yijing Yang, Wei Wang, C.-C. Jay Kuo, "Ensembles of Feedforward-designed Convolutional Neural Networks", in *International Conference on Image Processing*, 2019