NYCU Introduction to Machine Learning, Final Project

110550034, 孫承瑞

1. Environment details

1.1. Python version:

3.10.12 (The default version of Python on Google Colab)

1.2. Framework:

PyTorch

1.3. Hardware:

- For training: NVIDIA V100 GPU (provided by Google Colab **Pro**)
- For inferencing: NVIDIA T4 GPU (provided by Google Colab)

2. Implementation details

2.1. Model introduction:

Quoting from the paper *"Fine-grained Visual Classification with High-temperature Refinement and Background Suppression" (Chou, Kao & Lin, 2023)*, I employed the HERBS model to train my bird recognition classification model. The HERBS model features two crucial modules, the high-temperature refinement module and the background suppression module, for extracting discriminative features and suppressing background noise respectively. According to the paper, the proposed HERBS can be integrated into various backbones and improves the accuracy to 93% on datasets CUB200-2011 and NABirds.

2.2. Model architecture:

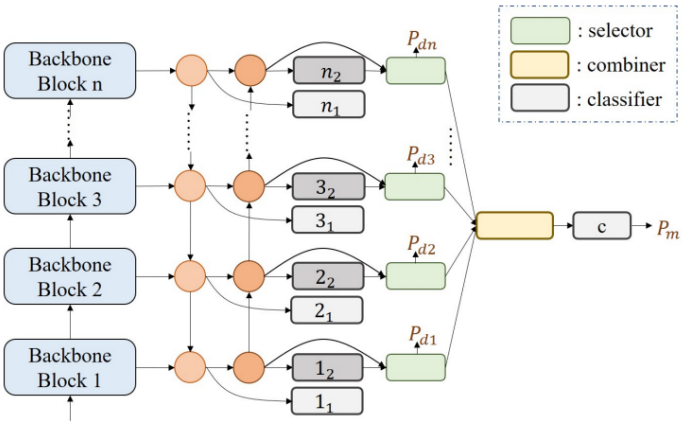


Fig. 2. The illustration of the model structure is shown, where the blue squares on the left represent the backbone blocks, which could be either Convolution-based or Transformer-based. The circles in the middle part denote the multi-scale feature fusion module, such as Feature Pyramid Network (FPN) or Path Aggregation (PA). The classifier, selector, and combiner on the right side depict the HERBS module.

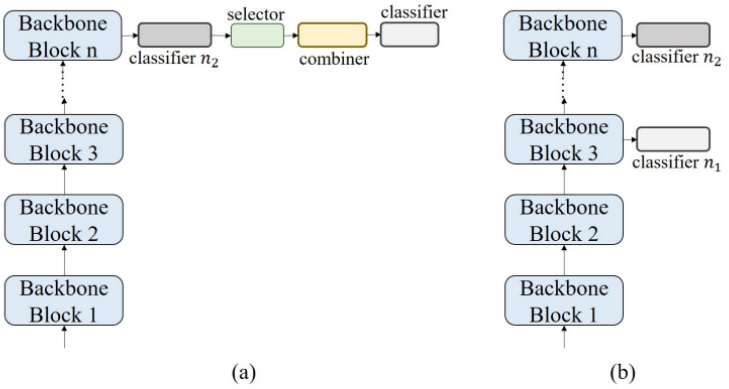


Fig. 3. Illustration of the structure of (a) basic background suppression module and (b) basic high-temperature refinement module.

In Fig. 2, the network is composed of the backbones, the top-down and bottom-up features fusion module, and the HERBS. HERBS contains two modules: the background suppression (BS) module and the high-temperature refinement module. In my model, I used **Swin Transformer Large** with 4 layers and **pretrained weights on ImageNet-22k** as the backbone and **FPN** as the feature fusion module.

In Fig. 3(a), the basic BS module is added to the output of the final block to partition the feature map into foreground and background. In Fig. 3(b), the basic high-temperature refinement module is applied to the last two blocks and enables classifier n_1 to discover broader areas in the earlier layer and classifier n_2 to focus on learning fine-grained and discriminative features in the later layer.

2.3. Hyperparameters:

I set the **learning rate to 0.0005** with cosine decay, and the model is trained for a total of **100 epochs** with the **batch size of 8**. I used **SGD** as the optimizer with the **weight decay of 0.0003**. The more detailed settings can be found in “training/configs/config.yaml”.

2.4. Training strategy:

For the provided dataset, I split it into **80%** training data and **20%** validation data, evenly distributed across each class. Since I employ the Swin Transformer, the input image should be 384×384. Therefore, data augmentation techniques such as RandomCrop, RandomHorizontalFlip, RandomGaussianBlur, and Normalization are applied during training, while CenterCrop and Normalization are utilized during validation.

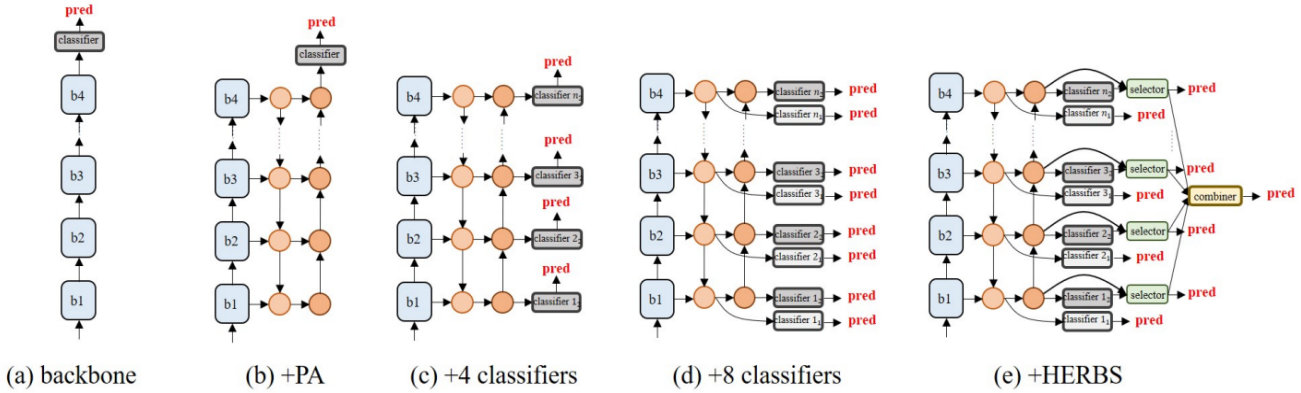
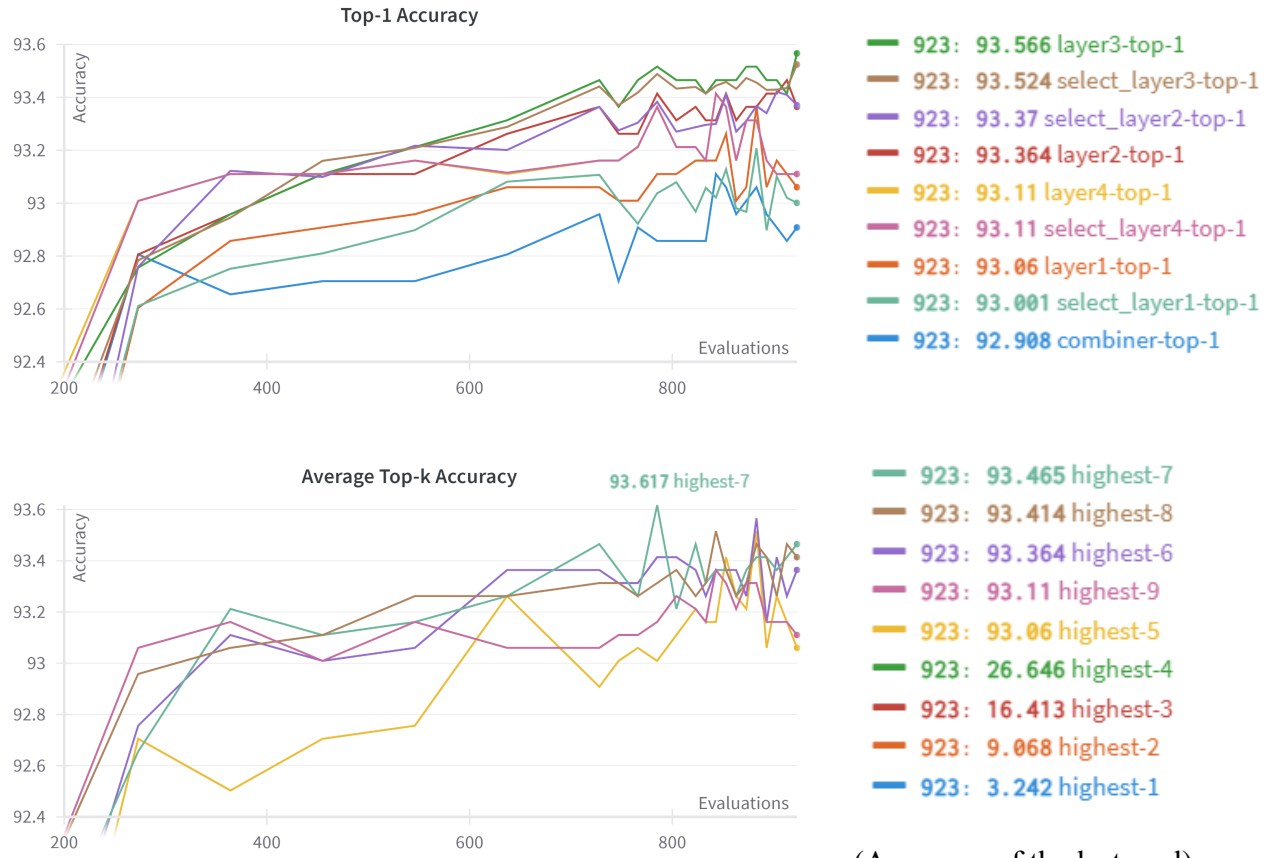


Fig. 4. The structure of models, (a) original backbone, the blue box represent the backbone blokcs. (b) backbone + path aggregation module. (c) backbone + PA module with four classifiers on the last bottom-up path. (d) backbone + PA module with eight classifiers on the top-down and bottom-up path. (e) backbone + HERBS

3. Experimental results

3.1. Evaluation metrics:

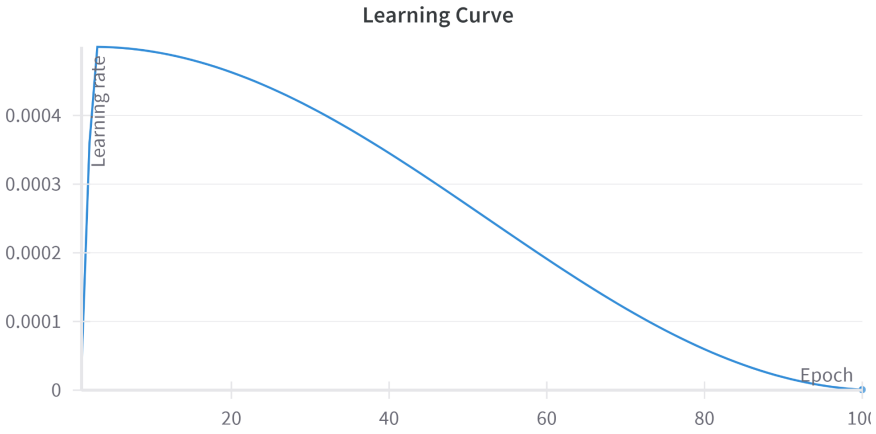
In Fig. 4(e), when employing the HERBS model, there are 9 predictions generated. During the evaluation of training, I calculated the **accuracy** of each **layer**, **combiner** and **highest-1 to highest-9** (highest-k means the Average Top-k Accuracy), and achieved the best accuracy of **93.617% from highest-7**.



(Accuracy of the last eval)

However, to get the final prediction, I **sum** all the classifier **softmax** outputs as the predicted result. After testing on the Kaggle competition, I got an accuracy of **91.3%** on the public leaderboard.

3.2. Learning curve:



3.3. Ablation Study:

To understand the impact of the **number of epochs** on accuracy, I trained the model for 80 and 100 epochs, and it took about 6 hours and 8.5 hours respectively. The result in the following table shows that the model trained for **100 epochs** has **0.9% higher** accuracy.

Epochs	80	**100**
Accuracy (%)	90.4	**91.3**

On the other hand, I wanted to compare the performance of the **HERBS model** to the **backbone**, so I removed the FPN, BS and refinement modules and then trained for 100 epochs again. It took about 6 hours this time. The result in the following table shows that the **HERBS model** has **1.2% higher** accuracy than the backbone.

	Backbone	**HERBS model**
Swin Transformer Large	V	**V**
FPN		**V**
BS		**V**
Refinement		**V**
Accuracy (%)	90.1	**91.3**

In the paper, the authors also provided some results and comparisons with different methods, backbones or modules. In TABLE I, the accuracy of the HERBS model is the highest one among other methods. In TABLE II and III, using Swin Transformer Large as the backbone and applying HERBS modules achieves the highest accuracy.

Method	CUB-200-2011	NA-Birds
FFVT[35]	91.6	N/A
ViT-NeT[17]	91.7	N/A
TransFG[9]	91.7	90.8
IELT[38]	91.8	90.8
SIM-Trans[29]	91.8	N/A
SAC[7]	91.8	N/A
CAP[1]	91.9	91.0
SR-GNN[2]	91,9	91.2
DCAL[51]	92.0	N/A
MetaFormer[4]	92.4	92.7
HERBS	**93.1**	**93.0**

TABLE I
COMPARISON OF TOP-1 ACCURACY(%) WITH STATE-OF-THE-ART METHODS ON THE TWO BENCHMARKS, CUB-200-2011 AND NA-BIRDS.

Module			Backbone	
PA	Refinement	BS	Swin-Base	Swin-Large
			91.3	92.0
✓			91.9(+0.6)	92.5(+0.5)
	✓		91.5(+0.2)	92.3(+0.3)
		✓	91.8(+0.5)	92.4(+0.4)
✓	✓	✓	92.3(+1.0)	93.1(+1.1)

TABLE II
COMPARISON OF TOP-1 ACCURACY(%) ON CUB-200-2011 WITH DIFFERENT MODULE ADDED TO SWIN TRANSFORMER.

Module			Backbone
PA	Refinement	BS	ResNet-50
			88.2
✓			88.6(+0.4)
	✓		88.7(+0.5)
		✓	88.4(+0.2)
✓	✓	✓	89.8(+1.6)

TABLE III
COMPARISON OF TOP-1 ACCURACY(%) ON CUB-200-2011 WITH DIFFERENT MODULE ADDED TO RESNET-50.

4. Conclusion

In this paper, HERBS, featuring the Background Suppression (BS) module and the high-temperature refinement module, are easily applicable to popular backbone networks and can achieve high accuracy up to 93% by effectively filtering out background noise and focusing on discriminative features while maintaining a proper attention area scale. Overall, the proposed HERBS provides a promising solution to improve the performance of fine-grained visual classification tasks.

5. References

- Po-Yung Chou, Yu-Yung Kao, and Cheng-Hung Lin. Fine-grained Visual Classification with High-temperature Refinement and Background Suppression. 2023 Referenced code: https://github.com/chou141253/FGVC-HERBS
- Weights & Biases, helping me plot the charts: https://wandb.ai/
- My GitHub repository: https://github.com/Rebeccasun31/2023-ML-final