Building Model

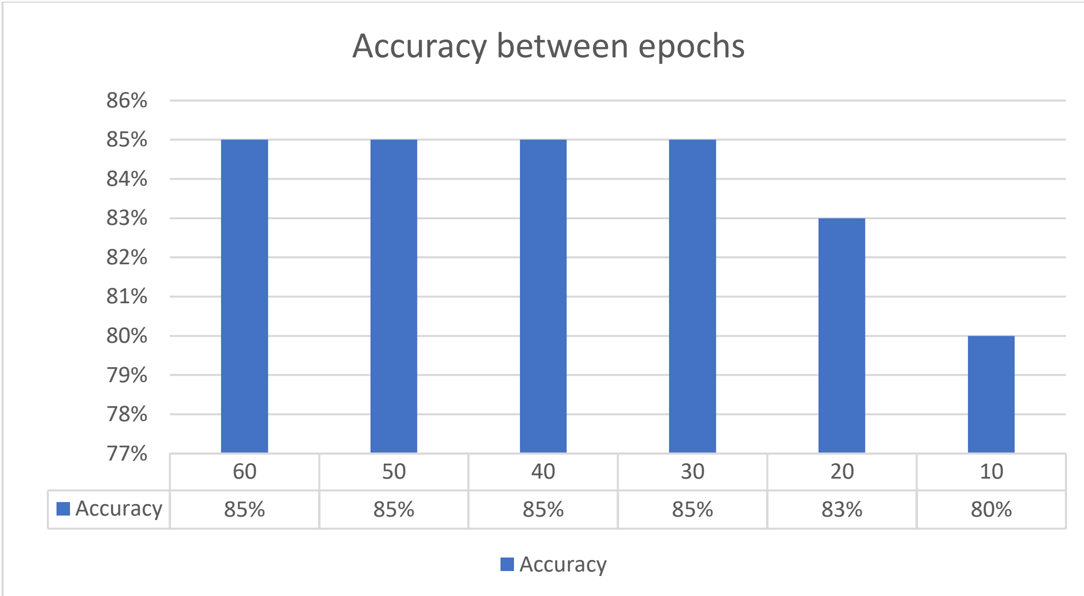
1. Choose model for classification

a) Base model

The selected model is ResNet-18 because it is simple and brings high efficiency to tasks with few classes. This was also demonstrated in the experimentation process with this problem.

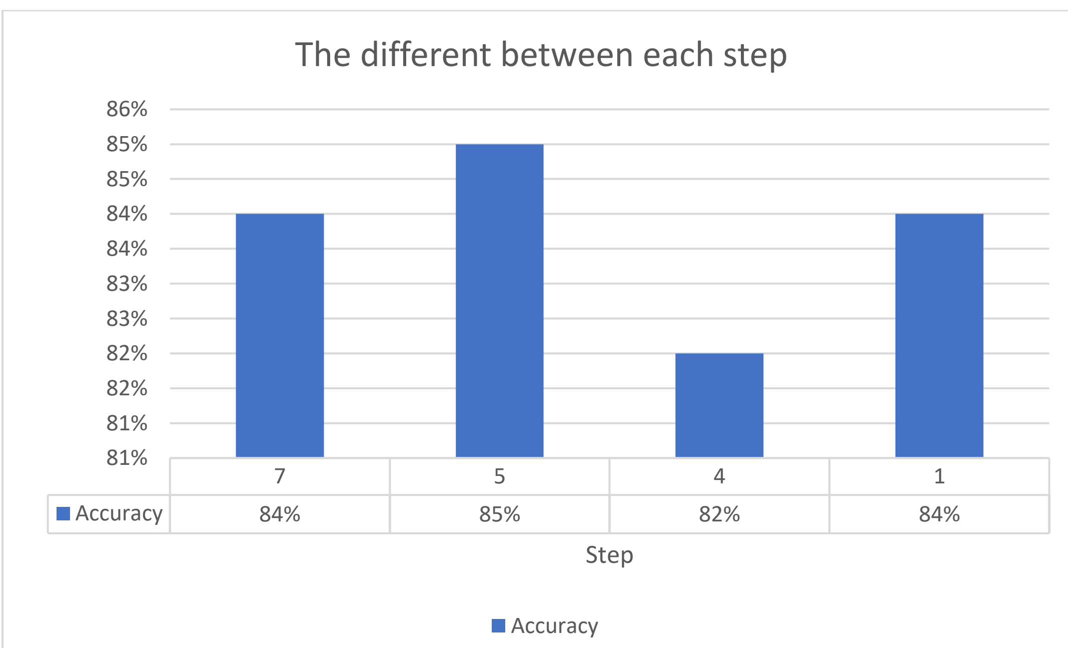
Model	Accuracy
Resnet - 18	**85%**
Resnet - 50	83%
Resnet -101	81%
Resnet -152	82%

The loss function used in this model is cross-entropy because it is the most basic loss function in classification, is easy to use, and still yields high effectiveness. The optimizer can choose between SGD momentum and AdamW. Firstly, let's analyze these optimizers. SGD momentum uses the principle of gradient descent to find the minimum weight and bias. This is an improved version when momentum is added to avoid getting stuck when finding the minimum weight. However, adding such momentum can also lead to passing through the point that needs to be reached. Therefore, this index needs to be reasonably appropriate. AdamW is a method that combines the advantages of RMSProp and SGD momentum. AdamW is also an improvement over the Adam method by adding weight decay to reduce the risk of overfitting. AdamW usually brings slightly higher accuracy than SGD momentum, but the results are often slower. However, in this problem, AdamW yielded better results (85% compared to 81%). Therefore, we choose AdamW as the optimizer and choose a learning rate of 1E-3 and weight decay of 1E-4 (these are the original values provided by the author of AdamW in https://arxiv.org/pdf/1711.05101v3.pdf). The batch size can be chosen from several different numbers, such as 4, 16, 32, and 64, but when experimenting, the results showed no significant difference. However, a batch size of 32 has a faster speed than other batch sizes. Choosing an appropriate number of epochs is also very important, so many epochs have been selected for testing.



After observing the training process, it was found that the accuracy did not change much after epoch 30. Therefore, epoch 30 was chosen as appropriate.

For the learning rate schedule, we chose the default gamma value of 0.1 and tested with different step values.

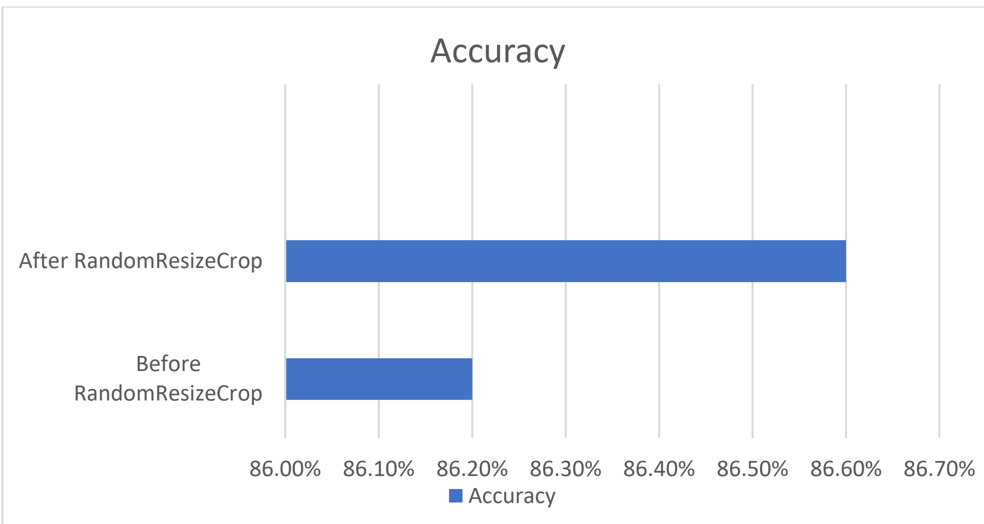


b) Model with data argumentation

To improve the accuracy of the model, adding supplemental data is crucial. The first augmentation that can be selected is resizing. Resizing can reduce parameters and increase focus on important image points.

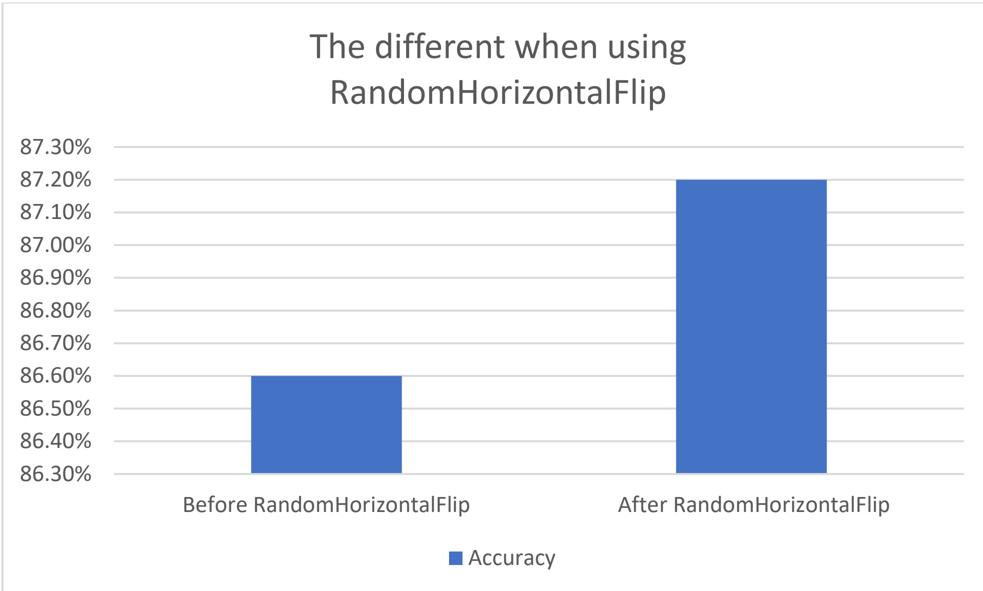
Resize	**Accuracy**
100(Default)	85.3%
96	**86.2%**
64	85.8%
32	85.6%

The next augmentation applied is RandomResizeCrop, which creates randomness when resizing images.



We can also apply RandomHorizontalFlip to increase the randomness of the data.

Changing the orientation of the image also yields good results in experimentation.



To increase the speed of training, we will convert the image data to tensors and normalize them. This may not significantly increase accuracy, but it has reduced training time from 1 hour 45 minutes to 1 hour 15 minutes. To increase accuracy and reduce overfitting, we will use a different argumentation method for validation, namely center crop. Using different argumentation techniques has increased the effectiveness of the training process from 87.2% to 88.7%.

In summary, we have a complete data augmentation as follows:

- Training: Resize (96), RandomResizeCrop (64), RandomHorizontalFlip, ToTensor, and Normalize.
- Validation: Resize (96), CenterCrop (64), ToTensor, and Normalize
- Test: Resize (96), ToTensor, and Normalize.

c) Data Validation

Having a validation set will create a more suitable dataset and increase the effectiveness of the training process. First, we have blur detection. By using the Laplacian method to calculate the focus index of the image and determining whether the image is blurry or not, Then, we will consider a reasonable threshold to remove blurry images. The selected threshold in this case is 400. The problem is that when using the resize method, the sharpness of the image will be increased if calculated by this method. Therefore, it is necessary to process directly with raw images. Next is Bright Adjustment, which uses the average calculation of color points on an image to determine whether the image is dark or light (128–255 is light, 0-127 is dark). Then we set a certain range to use methods such as bright or contrast adjustment to adjust the image to a reasonable brightness level (115–200). In summary, with the use of a validation set, the remaining number of images is 8K for training, 1.9K for validation, and 1K for testing. When running with the previously achieved checkpoint of 88.7%, we achieved 93.7% with the validation data.

d) The completed model