

Exercise 3: Convolutional Networks

Students: *Mehran Jeelani (2581358), Beste Ekmen (2572855), Md Akib Shahriar Khan (2581361)*

Course: *High-Level Computer Vision*
Due date: *June 7th, 2020*

Question 1: Implement Convolutional Network (10 points)

(a) Train the model and report the training and validation accuracy.

Answer.

- Training accuracy at the end of 20 epochs : 9.87%
- Validation accuracy at the end of 20 epochs: 8.7%

(b) Report the number of parameters for the described model.

Answer. The total number of trainable parameters for our model without batch norm are **7,941,130**. With batch norm we have **7,942,410**

(c) Compare the filters before and after training. Do you see any patterns?

Answer. The filters after training appear dark. This is because we are facing dying Relu problem and as such the accuracy is abysmally low.

Question 2: Improve training of Convolutional Networks (10 points)

(a) Compare the loss curves and accuracy using batch normalization to its counterpart in Q1.a.

Answer. Accuracy using batch norm for 20 epochs:

- Train: 71.57%
- Validation: 40.2%



(b) Increase the training epochs to 50 in Q1.a and Q2.a, and compare the best model and latest model on the training set.

Answer. Early stopping with Batch Norm:

	Fully Trained Model	Early Stopped Model
Train	100%	84.43%
Val	49.2%	60.4%
Test	47.6%	59.1%

Early Stopping without Batch Norm:

	Fully Trained Model	Early Stopped Model
Train	9.99%	9.96%
Val	8.7%	11.9%
Test	8.9%	10%

Question 3: Improve generalization of Convolutional Networks (10 points)

(a) Discuss which augmentations work well for you in the report.

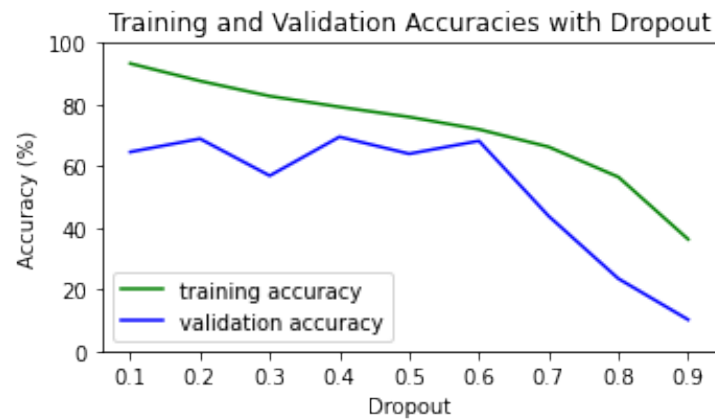
Answer. We used,

- `transforms.ColorJitter(hue=.05, saturation=.05)`
- `torchvision.transforms.RandomGrayscale(p=0.1)`

our validation accuracy increased from **40.2%** to **57.5%** with batchnorm additionally employed.

(b) Show the plot of training and validation accuracy for different values of dropout (0.1 - 0.9) in the report.

Answer.



We observe that with **drop probability = 0.4**, we get best validation accuracy of **69.5%**

Question 4: Use pretrained networks (10 points)

(a) Get familiar with gradient control mechanism in PyTorch and train the mentioned model.

Answer. We have trained the mentioned model in the question. Our accuracy results:

- Validation: 62.2%
- Test: 62%

(b) Train a baseline model where the same entire network is trained from scratch, without loading the ImageNet weights. Compare the two models training curves, validation and testing performance in the report.

Answer. Validation and test accuracy for the baseline model and the ImageNet pre-trained model:

	Baseline Model	ImageNet Pretrained Model
Val	86.3%	87%
Test	85.4%	88.4%

Based on our results, pretrained model converges faster than the baseline model and it shows a slightly better performance.

