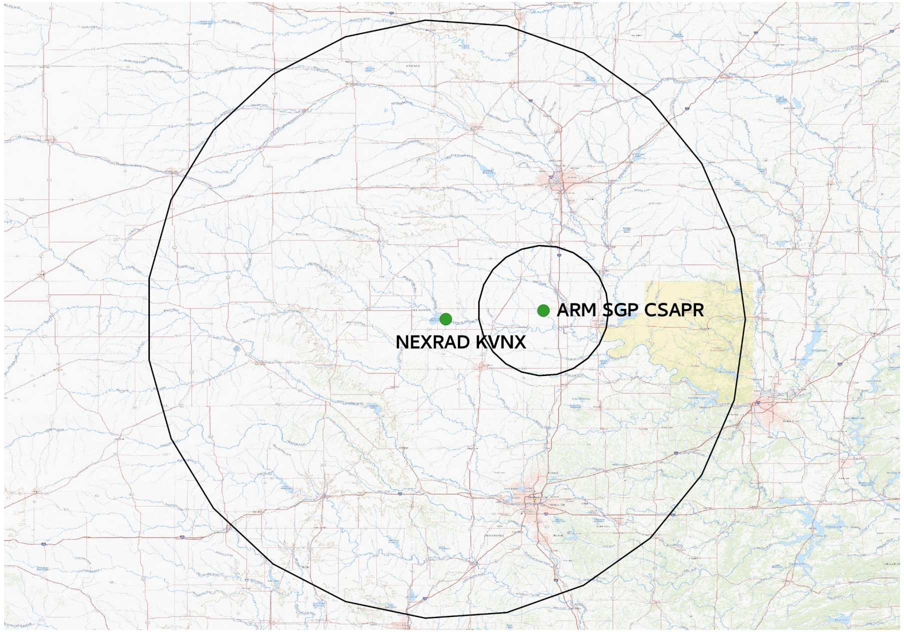
Deep Convolutional Neural Networks for Hydrometeor Classification using Dual Polarization Doppler Radars

Yuping Lu, Jitu Kumar

ARM Data Science and Integration, ORNL

06/10/2019

The Problem



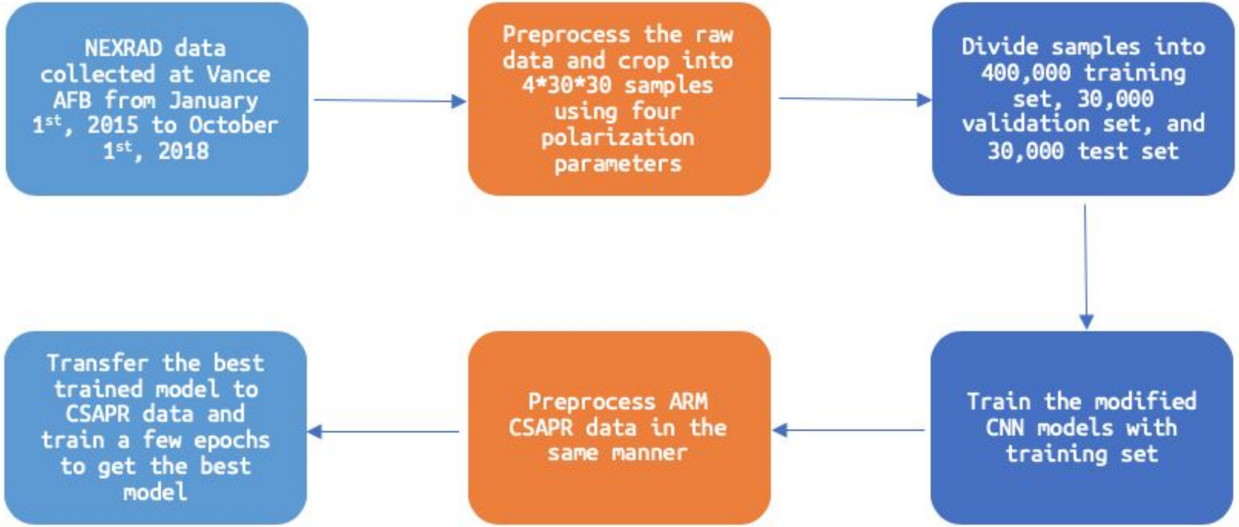
CSAPR: C-band scanning ARM precipitation dual-polarization Doppler weather radar.

In addition to the first three Doppler moments (reflectivity, radial velocity, and spectra width), the C-SAPR also provides differential reflectivity, correlation coefficient, and specific differential phase. The dual-polarization variables enable the provision of rainfall rate estimates and identification of precipitation types.

Four variables, horizontal reflectivity (Z_H), differential reflectivity (Z_{DR}), correlation coefficient (ρ_{HV}) and specific differential phase (K_{DP}), are used to build the model.

Data have no labels.

Training workflow



Training Environment

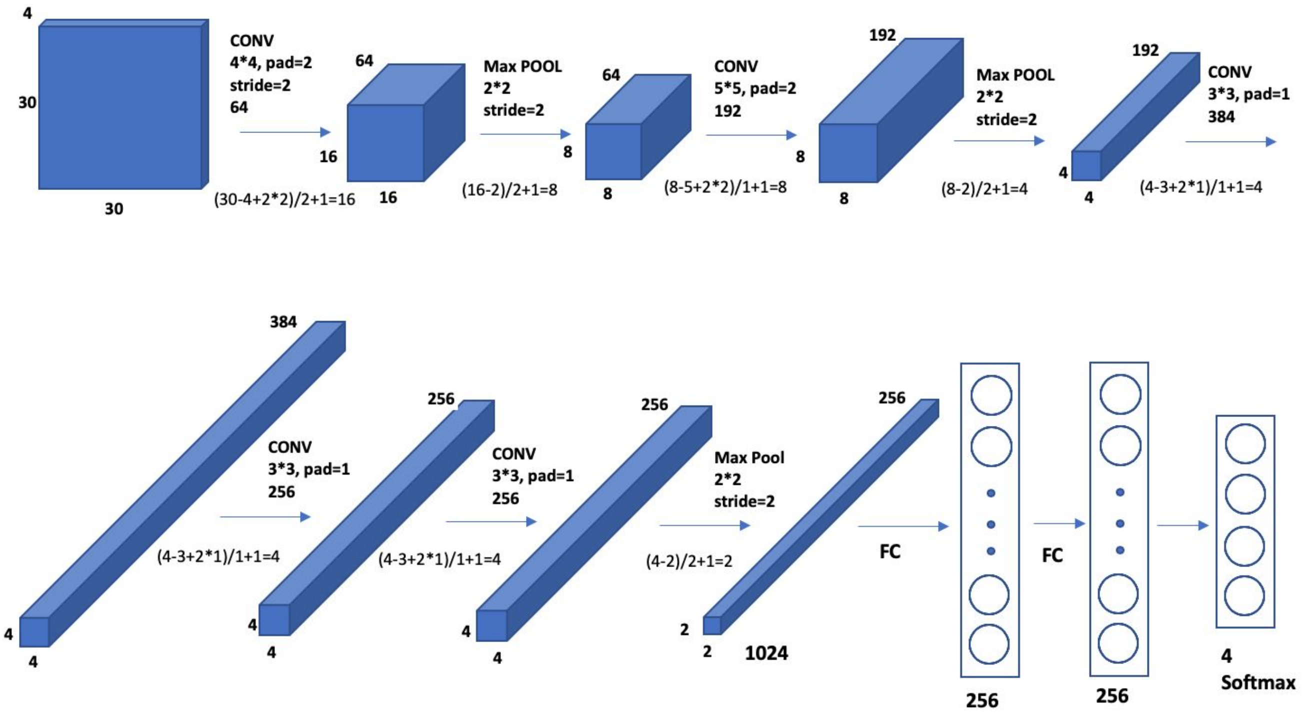
Preprocessing: ORNL CADES

Training: NVIDIA DGX Station + PyTorch

Programming language: Python

Other tools: JupyterLab, GitHub

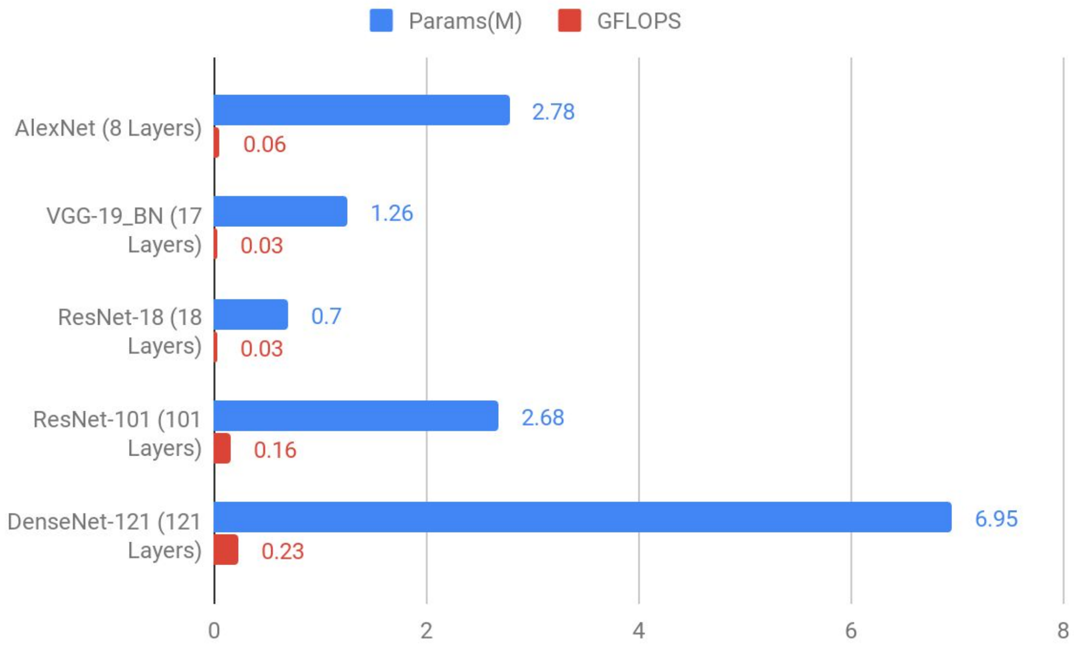
AlexNet architecture for NEXRAD



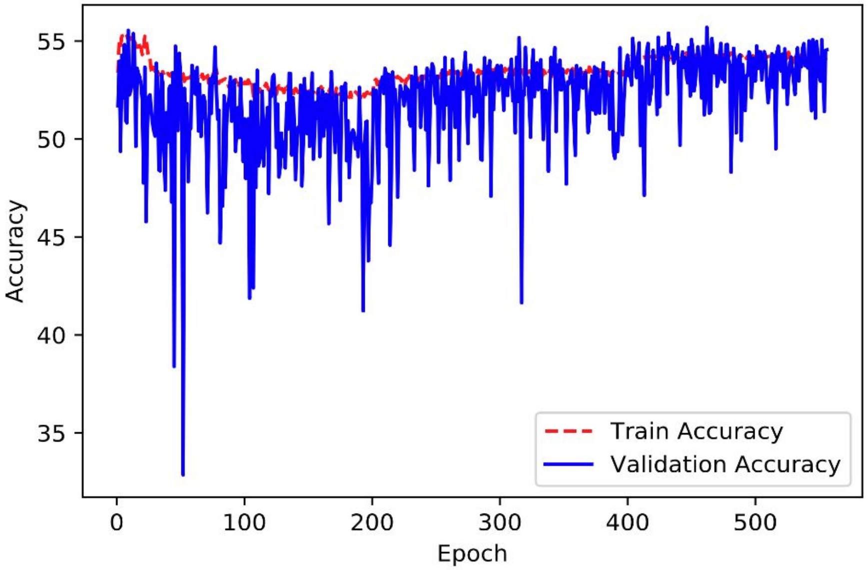
ResNet architectures for NEXRAD, number of parameters and GFLOPS

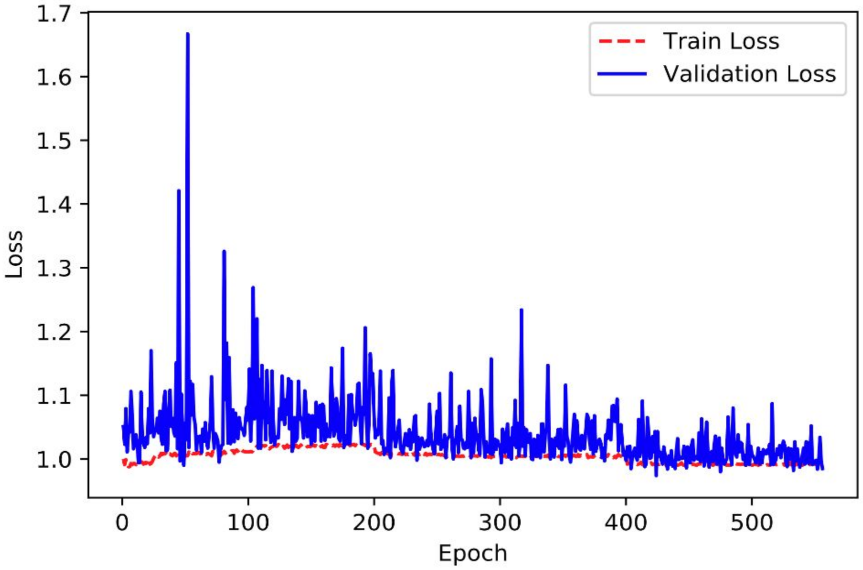
layer name	output size	18-layer	101-layer
conv1	30×30	3×3, 16, stride 1	
	30×30	3×3 max pool, stride 1	
conv2_x	30×30	$\begin{bmatrix} 3 \times 3, 16 \\ 3 \times 3, 16 \end{bmatrix} \times 2$	$\begin{bmatrix} 1 \times 1, 16 \\ 3 \times 3, 16 \\ 1 \times 1, 64 \end{bmatrix} \times 3$
conv3_x	15×15	$\begin{bmatrix} 3 \times 3, 32 \\ 3 \times 3, 32 \end{bmatrix} \times 2$	$\begin{bmatrix} 1 \times 1, 32 \\ 3 \times 3, 32 \\ 1 \times 1, 128 \end{bmatrix} \times 4$
conv4_x	8×8	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 2$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 23$
conv5_x	4×4	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 2$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 3$
	1×1	average pool, 4-d fc, softmax	
Params(M)		0.7	2.68
GFLOPS		0.03	0.16

Number of parameters (in millions) and GFLOPS of each model

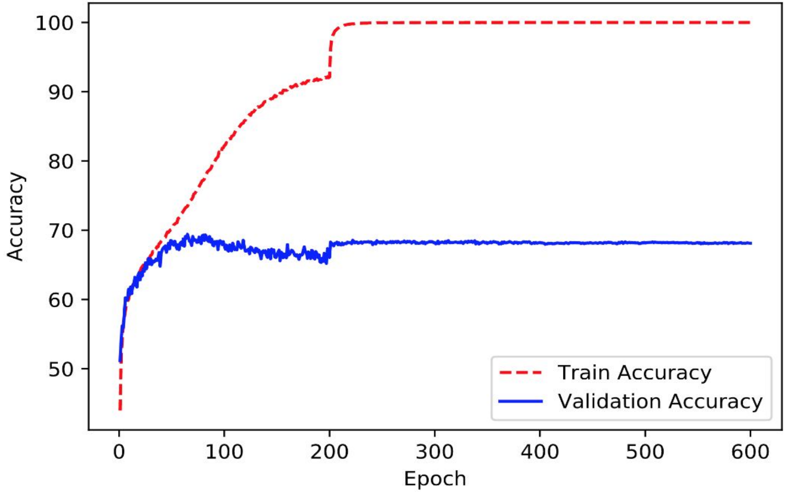


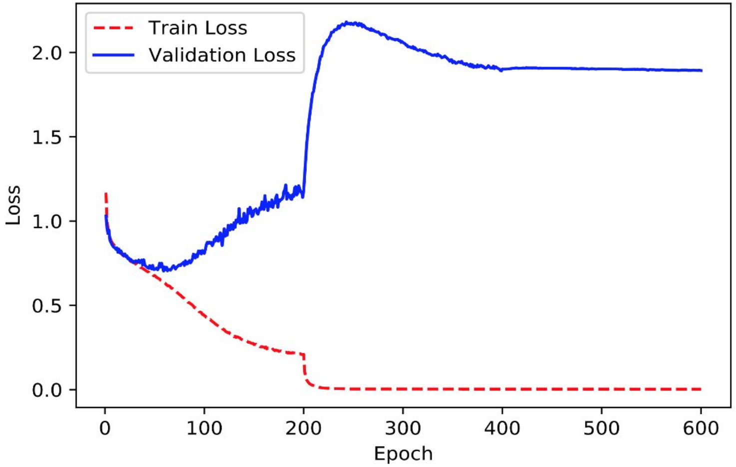
Not Learning Example



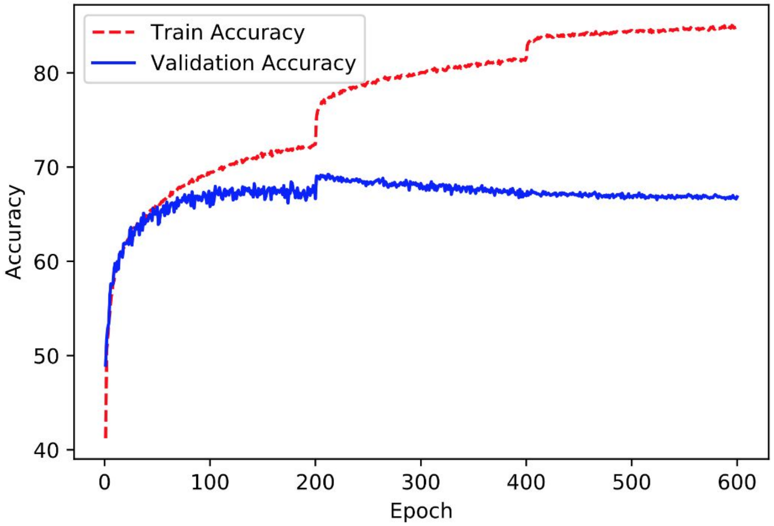


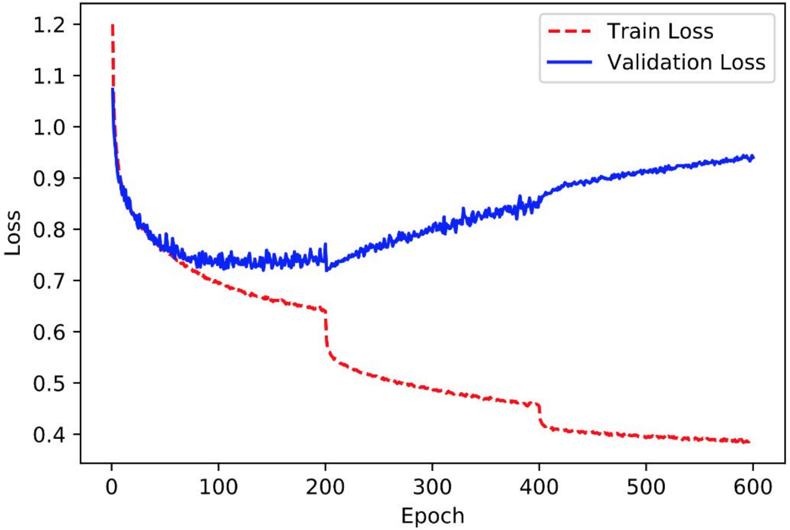
Overfitting Example 1



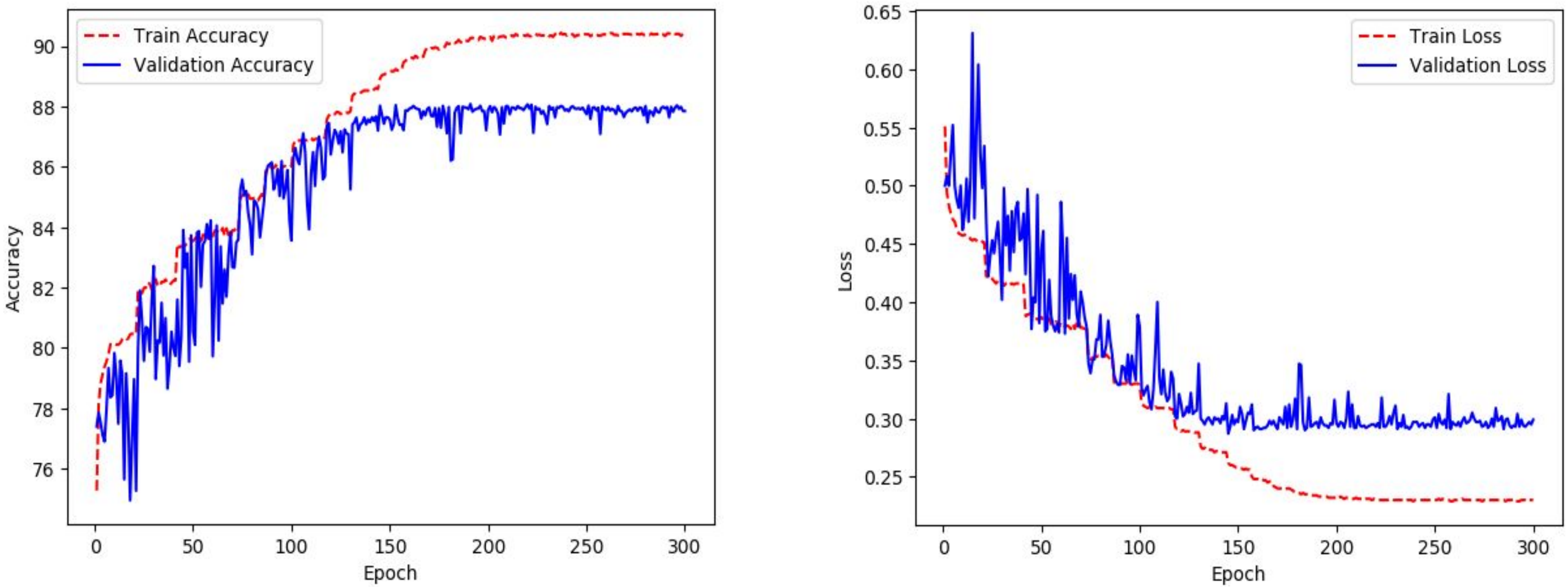


Overfitting Example 2





Overfitting Example 3



Reduce Overfitting

Fetch more data

Try some new approach in Neural Network

Change batch size

Try Regularization

...

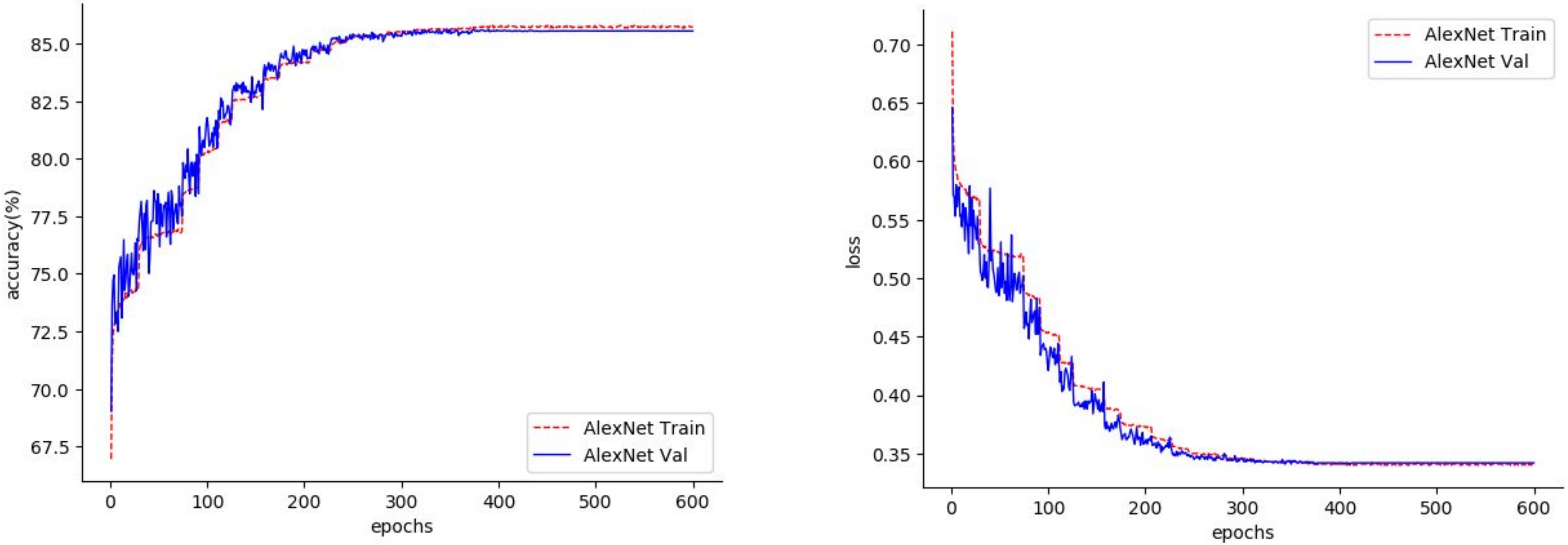
Reduce Overfitting

Add more data to the training.

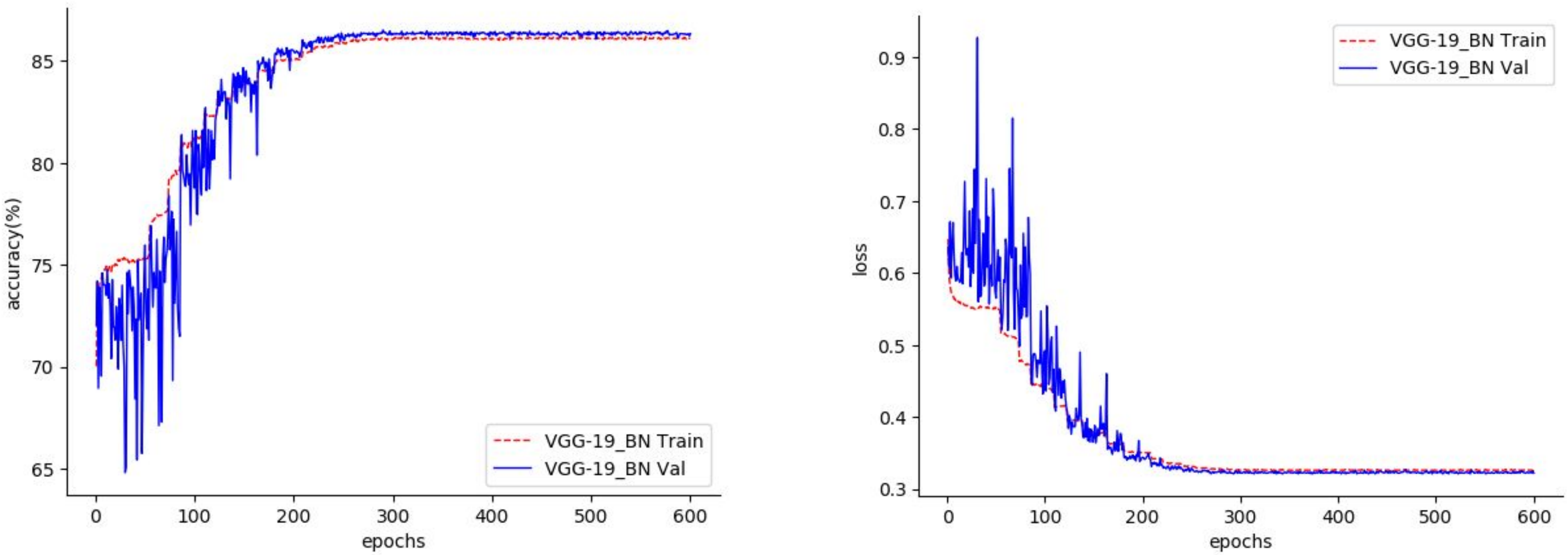
Reduce input size from 4*60*60 to 4*30*30.

Data augmentation.

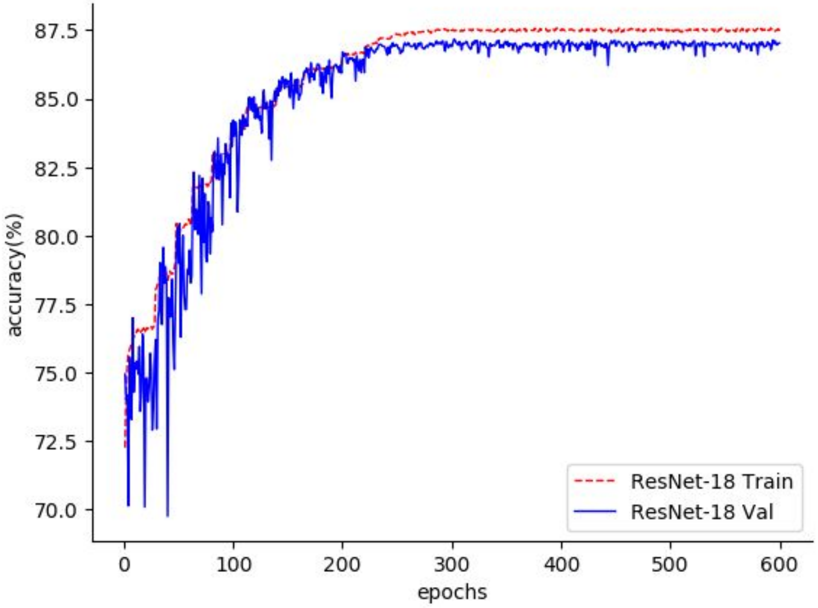
Hyperparameter Tuning.

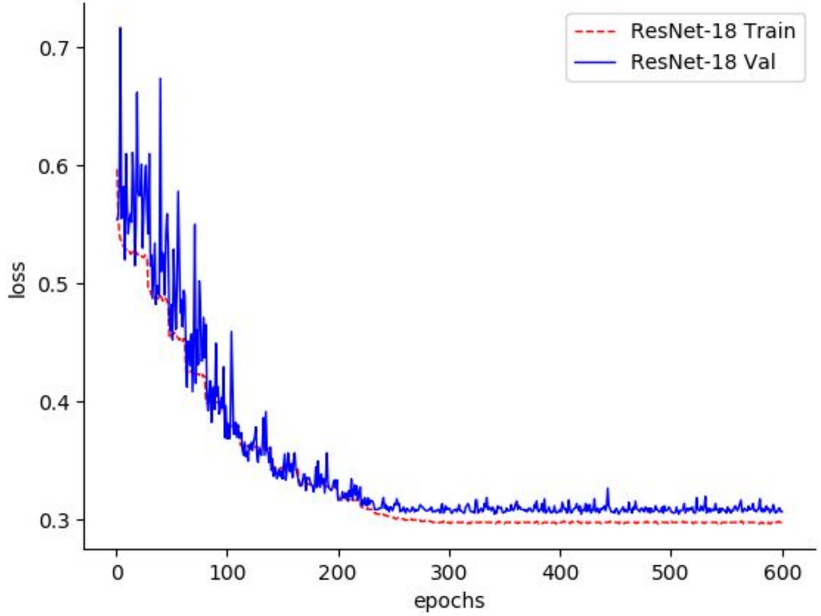


RandomCrop(), RandomHorizontalFlip(), RandomVerticalFlip(), batch-size = 256, epochs = 600, lr = 0.1, momentum = 0.9, weight-decay = 1e-3, dlr-decay = 0.5

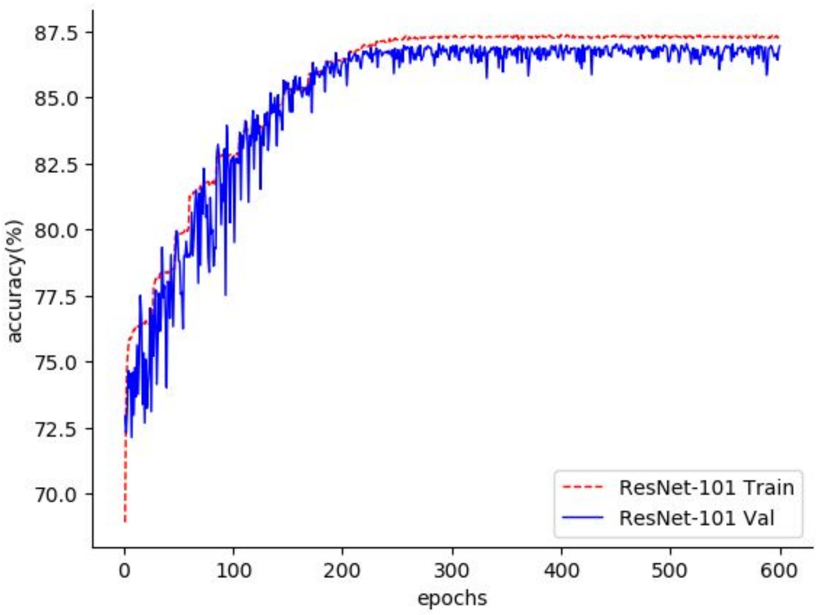


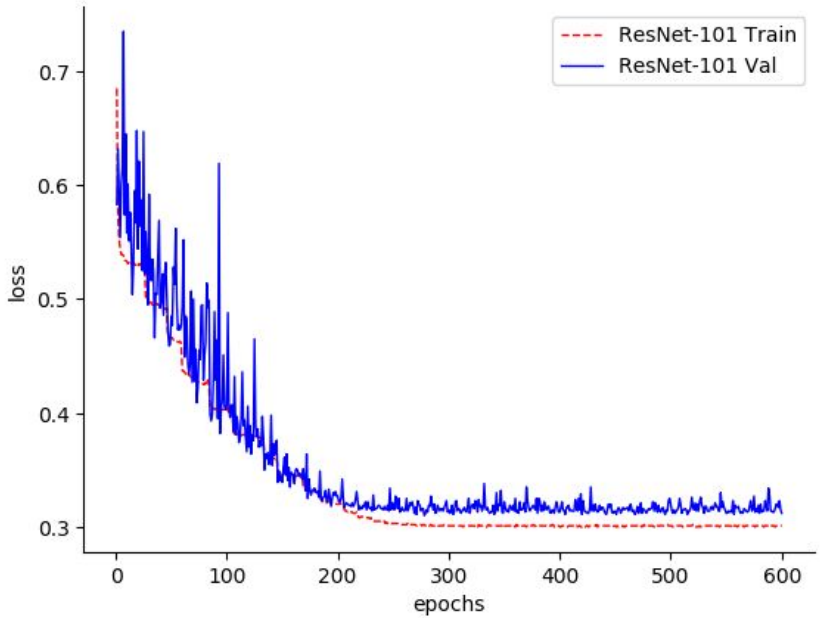
RandomCrop(), RandomHorizontalFlip(), RandomVerticalFlip(), batch-size = 256, epochs = 600, lr = 0.1, momentum = 0.9, weight-decay = 1e-3, dlr-decay = 0.5



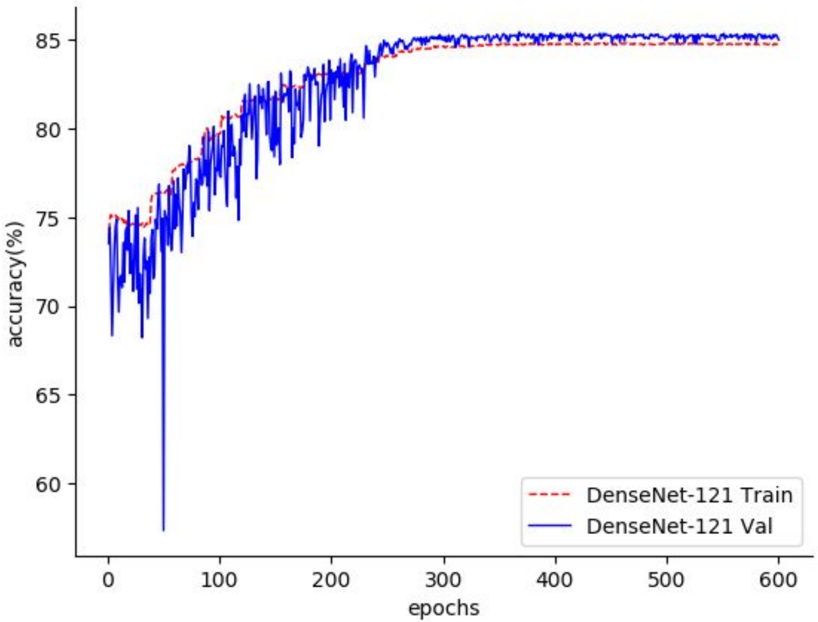


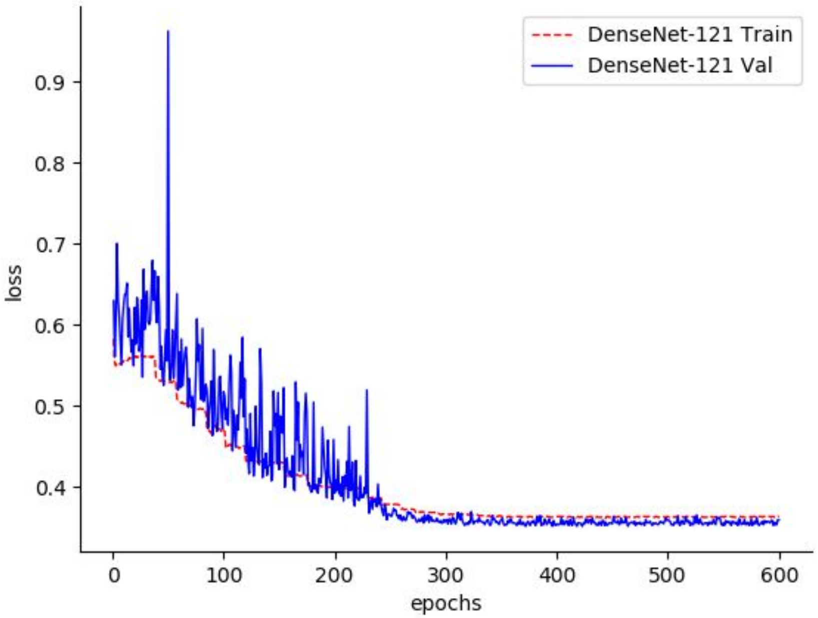
RandomCrop(), RandomHorizontalFlip(), RandomVerticalFlip(), batch-size = 256, epochs = 600, lr = 0.1, momentum = 0.9, weight-decay = 1e-3, dlr-decay = 0.5



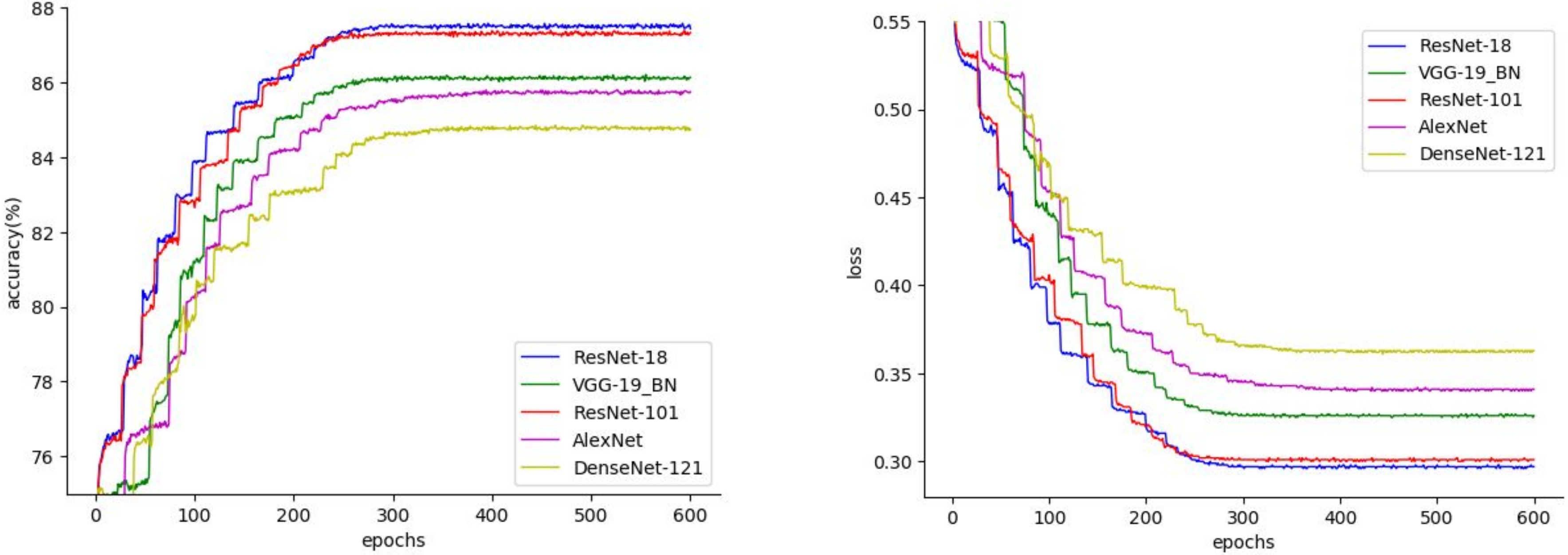


RandomCrop(), RandomHorizontalFlip(), RandomVerticalFlip(), batch-size = 256, epochs = 600, lr = 0.1, momentum = 0.9, weight-decay = 1e-3, dlr-decay = 0.5

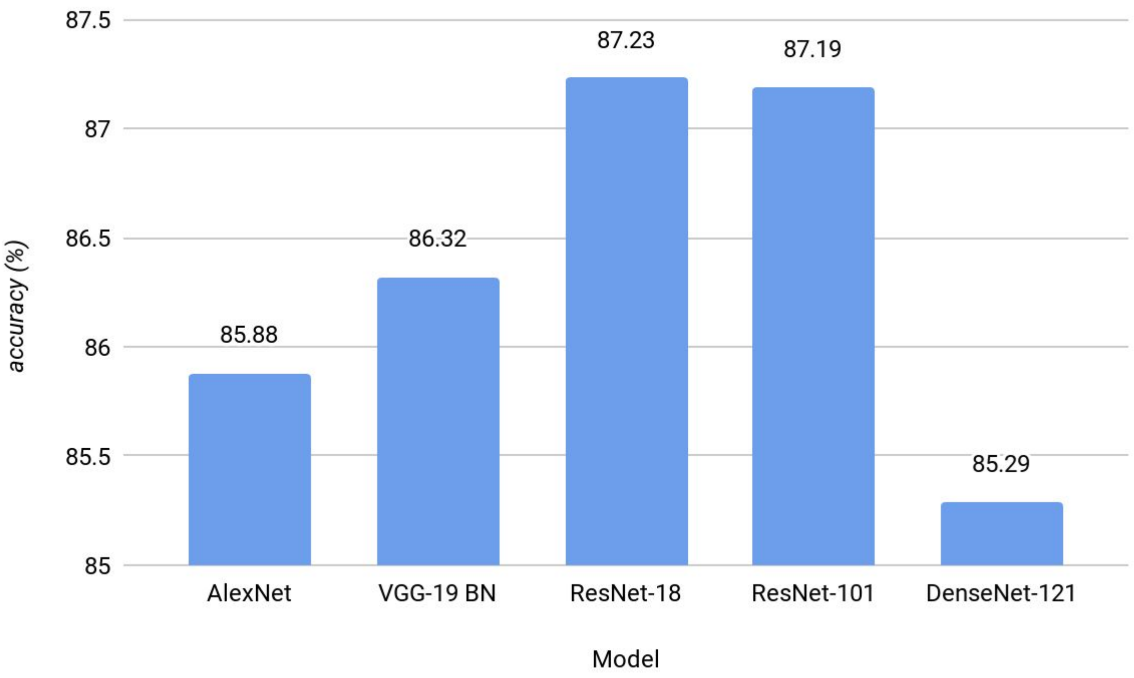


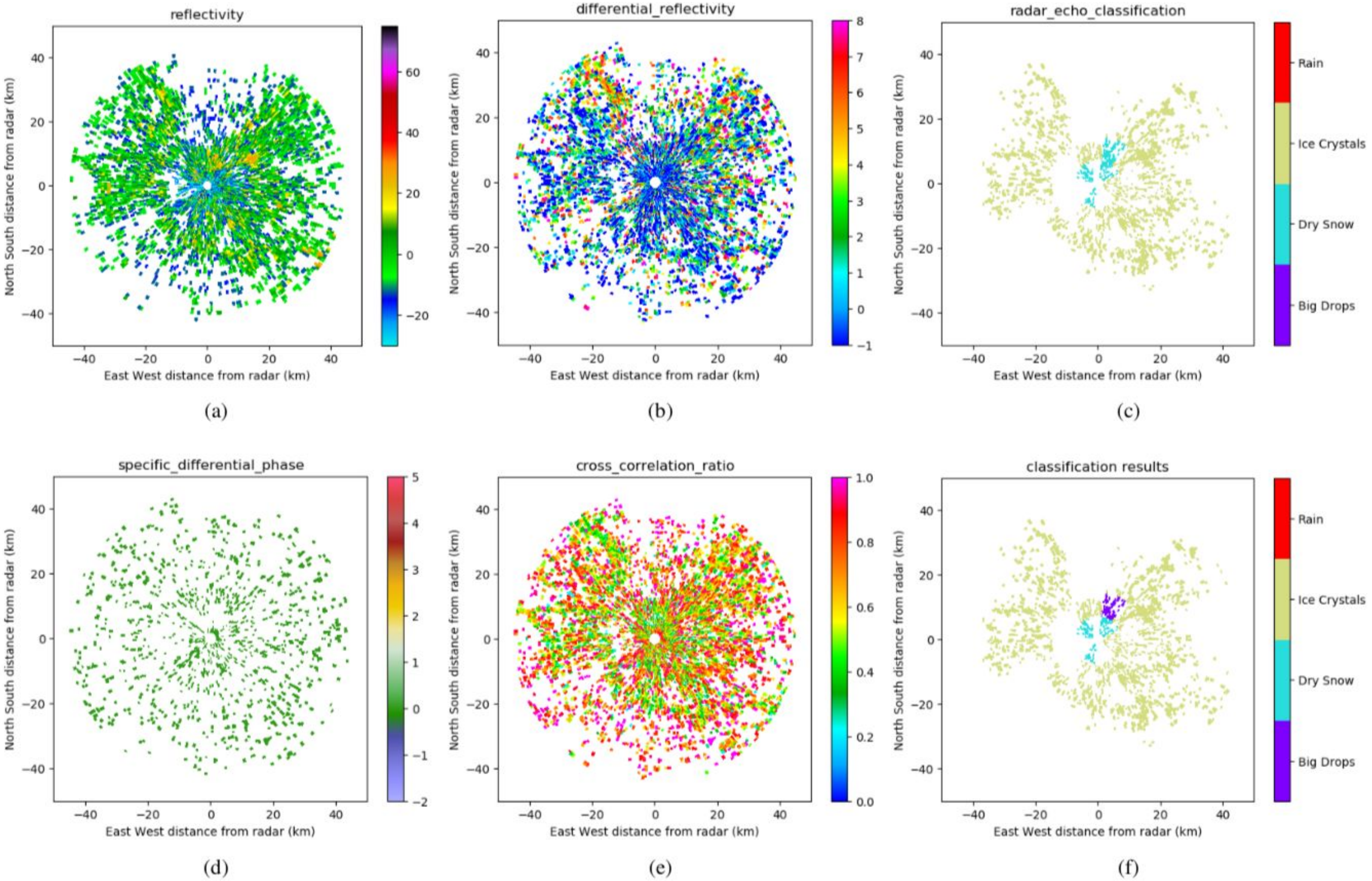


RandomCrop(), RandomHorizontalFlip(), RandomVerticalFlip(), batch-size = 256, epochs = 600, lr = 0.1, momentum = 0.9, weight-decay = 2e-3, dlr-decay = 0.5


accuracy(%)
epochs
88
86
84
82
80
78
76
0
100
200
300
400
500
600
ResNet-18
VGG-19_BN
ResNet-101
AlexNet
DenseNet-121
loss
epochs
0.55
0.50
0.45
0.40
0.35
0.30
0
100
200
300
400
500
600
ResNet-18
VGG-19_BN
ResNet-101
AlexNet
DenseNet-121

Accuracy of each model on NEXRAD test set




reflectivity
differential_reflectivity
radar_echo_classification
specific_differential_phase
cross_correlation_ratio
classification results
North South distance from radar (km)
East West distance from radar (km)
Rain
Ice Crystals
Dry Snow
Big Drops
(a)
(b)
(c)
(d)
(e)
(f)

The Next Step

```
203_0_60.csv IC
54_90_0.csv RA
103_120_60.csv IC
206_210_30.csv IC
98_300_30.csv RA
218_30_30.csv RA
101_60_0.csv IC
130_240_60.csv IC
79_60_0.csv RA
96_180_0.csv IC
2_300_60.csv IC
227_60_30.csv RA
34_210_0.csv RA
226_330_0.csv IC
12_0_30.csv RA
115_120_0.csv IC
238_240_60.csv IC
229_210_60.csv DS
6_30_0.csv RA
6_210_60.csv IC
133_180_30.csv IC
29_90_30.csv IC
```

Use trained model to classify ARM CSAPR data.

Cooperate with radar experts to verify the classification results.

Finally to create Value-added products (VAPs) with the trained models.

We also have a poster, you can find more during the poster session.
Thanks!