Deeply-Fused Branchy Network

- Small-but-complete side branches
  - fewer layers, fewer output channels, faster inference
  - fully formed architectures resembling the main stream
  - make branches still achieve good enough accuracy
  - increase the ratio of exit samples at earlier branches
- Sequential decision making
  - early exit for easy-to-discriminate samples
  - save computation time
- Collaborative decision making
  - make probability fusion for hard-to-discriminate samples
  - improve accuracy

Forward Inference

Inference Path

- Right stream ➔ Left stream ➔ Central main stream

Time-Accuracy Trade-off

- For time critical application, simply set the exit threshold to be 0 at the 1st earliest branch
- For accuracy critical mission, set the exit thresholds of earlier branches to be higher (e.g., 0.99)
- Exit thresholds provide the flexibility of controlling the trade-off between computation time and accuracy

Approach to Training Side Branches

- Firstly, train the main stream net from scratch
- Secondly, load the already-trained main stream model to fine-tune its branchy sub-nets

Joint Optimization During Training

Experiment Setup

- Basically CIFAR-10 and CIFAR-100 use the same DFB-Net architecture
- Apply Dropout or not
  - CIFAR-10 ➔ No
  - CIFAR-100 ➔ Yes
- Down-sampling method
  - CIFAR-10 ➔ Use Conv_1x1 with Stride 2
  - CIFAR-100 ➔ Use 2x2 Ave-Pool
- Apply scale and aspect ratio data augmentation
- Use GTX-1080, CUDA 8.0, cuDNN 5.1 for inference

Summary

- Our CIFAR-10 baseline model achieves state-of-the-art result with 3.23% error rate
- On CIFAR-10, our branchy network with fusion
  - achieve state-of-the-art result with 3.07% error rate
  - got 3.0x speedup while better than baseline model
- On CIFAR-100, our branchy networks with fusion
  - achieve state-of-the-art result with 16.01% error rate
  - got 2.75x speedup while better than baseline model