X-ray image body part clustering using deep convolutional neural network

Sungbin Choi

Seoul National Univ., South Korea

Introduction

Task

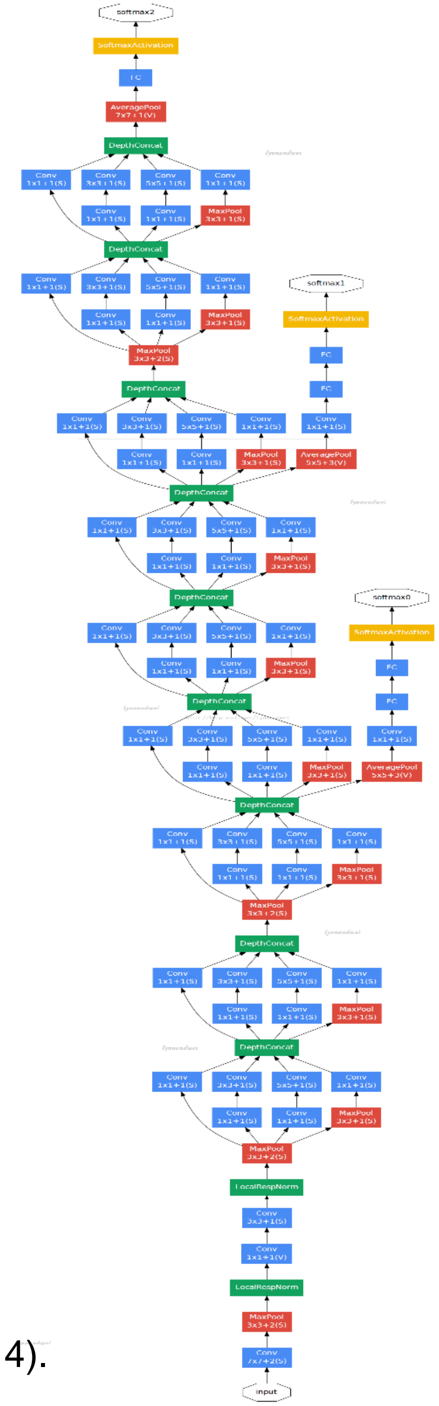
- Given digital x-ray images of various body parts,
  clustering images into four different body parts:
  head-neck, upper-limb, body and lower-limb

Method

- Multi-class image classification
- Simple application of pretrained deep convolutional neural network
- Finetune from GoogLeNet which is pretrained on ImageNet dataset
- Softmax loss function

GoogLeNet

- Deep convolutional neural network
- Inception module
  - Network in Network
- Pretrained on ImageNet database
  - 1000 categories
    - e.g., Zebra, Mountain bike, Orange



Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... & Rabinovich, A. (2014). Going deeper with convolutions. *arXiv preprint arXiv:1409.4842*.

Finetuning GoogLeNet

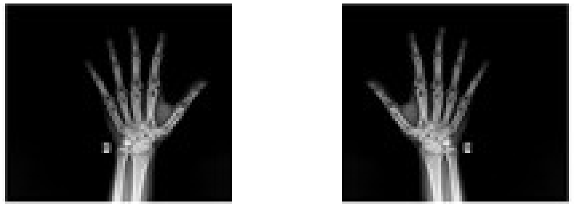
- Experimented with Caffe

- Initial learning rate 0.001
- Batch_size: 40

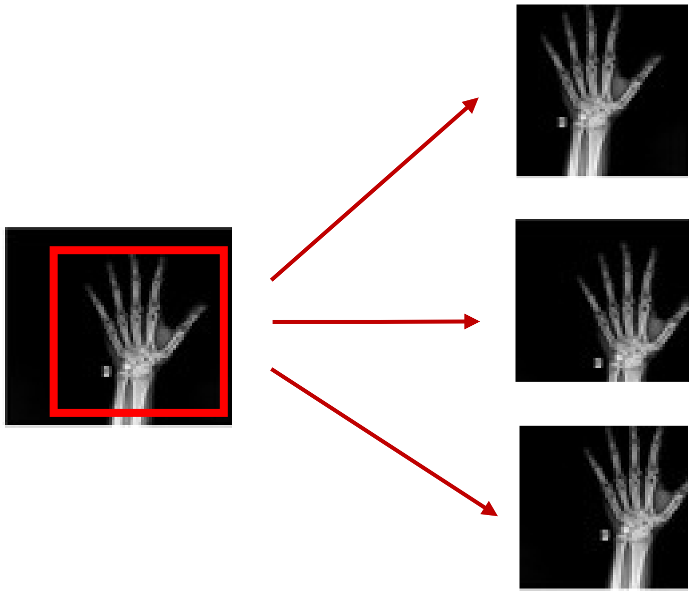
Data augmentation

- 90 degree random rotation (90’, 180’, 270’ and 360’)

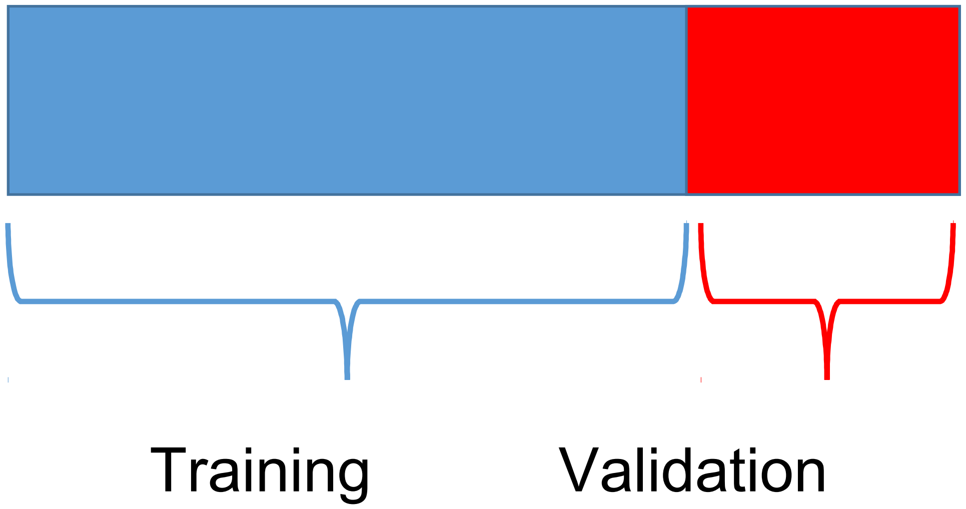
- Random mirroring (left-right flipping of image)



- Image resizing and random crop
  - 300x300 => 224x224

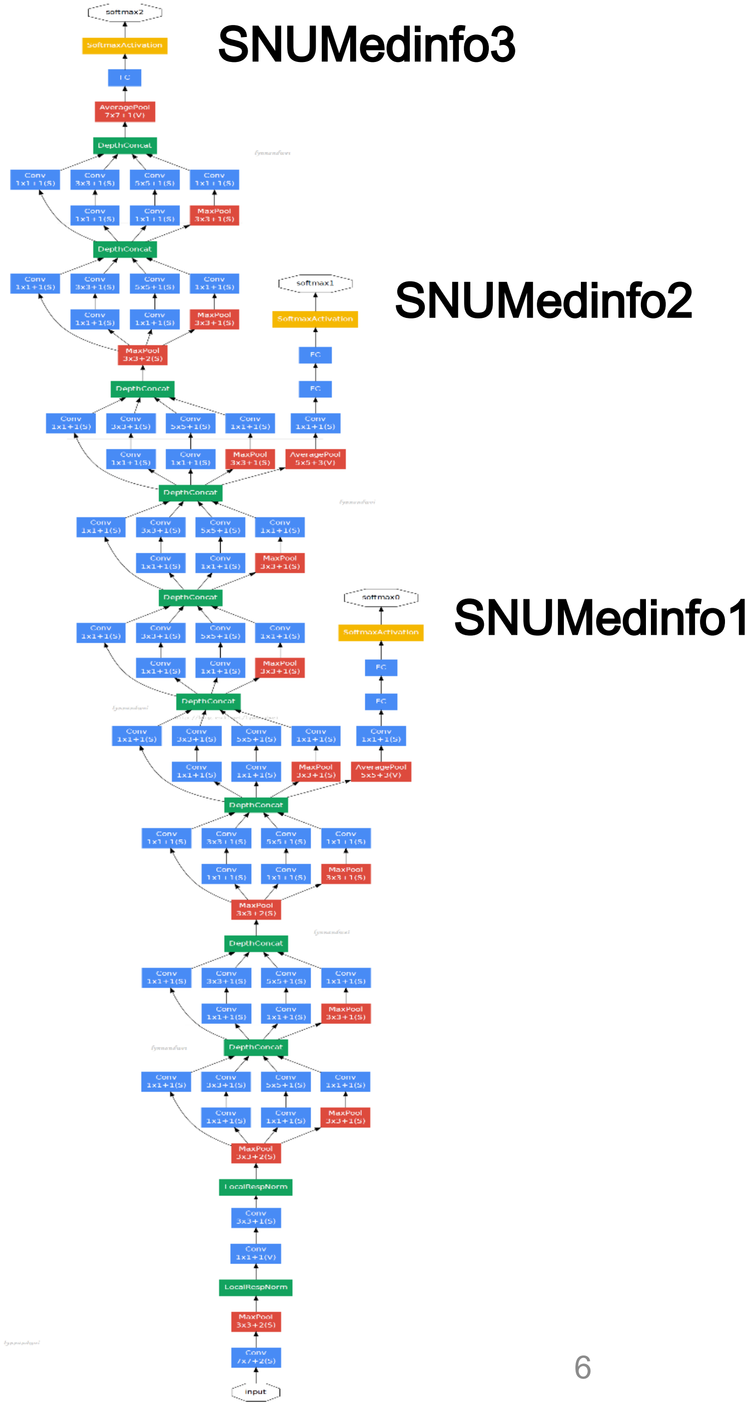


- 5-fold cross validation



- Per-model top-1 accuracy: 0.89~0.93

- Borda-fuse
  - Combine with rank information



Evaluation result

Name	Name of the Sumited file	Exact Match	Any Match	Hamming similary
IBM MMAFL	1431923404568__IBM_MRL_ef_mslbppyr_scodes_densemblbp_densesift_th.txt	0.752	0.864	0.863
SNUMedInfo	1431946260816__SNUMedinfo3.txt	0.709	0.856	0.895
SNUMedInfo	1431946236163__SNUMedinfo2.txt	0.699	0.844	0.890
IBM MMAFL	1431923215231__IBM_MRL_ef_mslbppyr_scodes_densemblbp_densesift_max.txt	0.695	0.840	0.889
IBM MMAFL	1431773613950__IBM_MRL_Sift_FV.txt	0.692	0.832	0.896
IBM MMAFL	1431773914574__IBM_MRL_SMO_edge_histogram_pyramid23.txt	0.692	0.732	0.755
IBM MMAFL	1431932028897__fuse_pred.txt	0.689	0.820	0.890
SNUMedInfo	1431946207517__SNUMedinfo1.txt	0.679	0.820	0.879
IBM MMAFL	1431922644645__IBM_MRL_mlhist_mslbp_pyr_max.txt	0.672	0.812	0.874
AmrZEGY	1431858638236__run1.txt	0.646	0.780	0.868
NLM	1431352031992__OpenI_SetA_5class.out	0.613	0.740	0.849
CASMIP	1431623161365__C_knnclassification.txt	0.606	0.732	0.843
CASMIP	1431621989456__svmclassification.txt	0.603	0.728	0.847
IBM MMAFL	1431773973109__IBM_MRL_SMO_lbp_histogram_pyramid23.txt	0.603	0.708	0.778
CASMIP	1431399984301__knnclassification.txt	0.599	0.724	0.847
IBM MMAFL	1431922924104__IBM_MRL_scodes_denseMBLBP_max.txt	0.599	0.724	0.838
NLM	1431634802808__OpenI_5class.out	0.593	0.716	0.842
CASMIP	1431623320464__TC_knnclassification.txt	0.589	0.712	0.842
CASMIP	1431623385130__TC_logclassification.txt	0.589	0.712	0.842
CASMIP	1431623447485__TC_logclassification.txt	0.589	0.712	0.842
CASMIP	1431622163959__logclassification.txt	0.573	0.692	0.833
CASMIP	1431622397898__dbnclassification.txt	0.573	0.692	0.833
CASMIP	1431623254773__TC_dbnclassification.txt	0.563	0.680	0.830

Discussion

- Finetuning from pretrained deep CNN model showed competitive performance
    - ImageNet dataset has different types of images from x-ray image
    - Small number (5k) of training set image
    - No domain knowledge is used
- We just let deep CNN model learn by themselves
- Augmented training data to give them more examples

Future work

- Scaling