Consolidated CV notes:

look up siamese triplet net

deeper network architectures can be difficult to train but have the potential to substantially

increase the network’s accuracy as they allow modeling mappings of very high complexity (SRGAN)

Classification

CNN (plankton)

**class imbalance? Prob. Because then its biased, has poor perf on smaller classes -> transfer learning** (pre-train with class normalized data [reduce big classes via random sampling, using a threshhold. Too big -> no bias reduction, too small -> lose specificity], then fine-tune with orig. i.e. train again on whole dataset) -> better acc than w/o or using other data augmentation techs. **Future? Sequential transfer learning w/ multi thresholds (learn to classify big datasets first, then smaller later)**

CNN (deepmole)

**expensive, time-consuming to feature extract/use expert features, also limited health data. Rather, use CNN to feature extract then classify by pre-training on normal data.** Use activations from earlier layers to make complete classifier because health data so different. Build custom classifiers for the last 3 (FC) layers. Then data augment w/ distortions. Custom classifiers use KNN w/ cosine dist because “more robust to outliers, used in other works”. **Future? Combine CNN w/ handcrafted features (??)**

CNN (Maxmin)

**using ReLu gets rid of negatives (i.e. negative info ignored, whether strong/weak) - > preserve this using 2 reLus to keep high + and - vals**

using CNNs for low-res and in weak supervision (used these days)

ReLu = standard activation func for classification, then max/avg pooling (which seem best). Relu zeroes all negative vals , done to “facilitate the exploitation of discriminative information by de-noising filter detections”. Can do PreLu (**check out**).

PCA (PCAnet)

**Emulate CNN’s processing layers thru PCA => easy to use, train and adapt. Also gives theoretical basis for deep stuff**

using CNNs for classification – depend on fine-tuning params. Not easy to generalize.

PCAnet = simple unsupervised CNN. 1. Cascading PCA on input. 2. Binary Hashing. 3. Block histograms. **Future? More sophisticated filters/stages so can handle extreme variability data sets (e.g. PASCAL VOC)**

CNNs

LBCNNs (Local Binary CNNs)

**Efficient approximation of a convolutional layer -> big computational savings, less prone to over-fitting**

training convolutional kernels end to end = expensive, large and over-fits limited data (cos so many params). Use binary weights instead of floats to save space (because can do binary convolutions (?)). but if complete binarization -> worse perf . Rather than learning to optimize conv filters, learn to optimize linear weights.

RCNN (Text Class)

**recurrent CNN for text classification without human-designed features**

n-gram (= no. words. 5 words = 5gram)

Bag of Words (BoW) models ignore sequence/context. Recursive Nns use tree (0n\*2) which is untractable. RNNs 0(n) but biased in favor of later words. -> bi-direction (Left and Right) recurrent CNN (get ‘left context’ and ‘right context’ in conv layer then max pool)

Recognition/Detection

Emotion Recognition using RCNN (Emote)

**Use CNN as input to an RNN. can’t use cnn alone cos then no temporal info**

use cnn’s regression layer for each frame as input to RNN

Deep Residual Learning

**more accurate = stack more layers? No. -> accuracy stagnation and degradation. NOT overfitting but higher training error. Raher – use identity mapping to overcome this and have super deep networks. Resnet = a fully convolutional network (FCN)**

rather than just stacking layers to hope they fit a desired underlying mapping, we explicitly let these layers fit a residual mapping. Do this using shortcut connections (ones that skip >=1 layers) -> no extra params and can have kaaaaaaak deep networks. **Prob: summing identity func and output of layer might cause info loss**

Densely Connected (DCNN)s

**Cnns better if connections close to input and close to output shorter. Instead of L layers having L connections, make all prev layers have connections to all subsequent layers and vice versa -> fewer params (don’t need to learn redundant feature-maps). Dense ones also deep supervised and less prone to overfitting cos self-regulate. Don’t go deeper/wider, do feature re-use.**

instead of creating a short path from early layers to late ones (a la resnet), connect *all* layers w/ matching feature map sizes directly [concatenate inputs]. Also prob with resnet is klomp params (each layer got own weights) but many of the layers do fokol.

Residual Attention Network (ran for img classification)

**CNNs + attention-aware features (thru *stacked* attention modules) -> more discriminative feature reps & higher obj classification accuracy and able to be super deep (also bc many attention modules -> can captture diff types of attention**

Segmentation

Conv Rand Walk nets

**Use convolutional random walk -> avoid low spatial resolution prob of FCNs w/ network that doesnt need post-processing & is compact and has standard training features**

fully conv networks good for semantic img segmentation but their ‘large receptive fields’ and lots of pooling -> low spatial resolution in their deepest layers -> blobby predicted segments & fixes to this prob dont capture semantic relationships btwn objects, and post-processing methods make harder to train and have klomp params

cervical cancer classifcation using multi-thresholding

**use mult-level thresholding then svm/knn to segment nucleus for cervical cancer diag. doesn’t need training and is super accurate**

VQA

**use giant human-given q’s and a’s dataset to get free-form vqas. Use deep LSTM (2 hidden layers) for qs and top-performing CNNs (VGG) for images.**

open-ended vqa combines klomp things : object and activity detection and recognition, image tagging/captioning, and common-sense and knowledge base reasoning. Often just ‘yay/nay’ but hard to get there, especially cos isnt MCQ. But also, could have synonymous answers/differing answers

Weak supervision (often just global img labels)

Much easier to global label images than use bounding boxes/annotations. So how to do obj recog etc. with just these global labels?

“WSL consists in designing accurate models able to pre-

dict detailed annotations, e.g. region localization, while

being trained from coarse labels, e.g. global image labels.” - GLSVM

WS Cascaded CNNs **IN PROG**

object detection usually done with strong supervision

usual approach to WSL = multiple instance learning (img= bag of instances[regions])

WS CNN for Img Clas (WILDCAT)

**Use Resnet ( less final layers ) +transfer layer (to learn multi class-related modalities (e.g. legs of dogs) + class-wise and spatial pooling on imgs w/ only image labels to learn discriminative localized visual features [to do obj recog and scene categorization]**

MO for CNNs is to train on large dataset e.g. ImageNet and then fine tune on target domain. BUT to do this you gotta detect objects/parts and context.

Deep CNNs are not super tolerant of scale and translation variations. How to turn MIL into global prediction? Maxpooling only takes most informative region (bad).

Gaze-Latent SVM

**Bias region selection using human gaze (eye tracker) cos almost zero cost and also quick.**

Use only gazes for training, and then only vis info for prediction

Vid Classification using min supervision

**Need only the ‘signs’ of features of the target class to classify**

approaches feature selection from task of discriminant linear classification.

“r is a weighted feature average with an ensemble response that is stronger on positives than on

negatives. For that, we want any feature f j to have expected value E P (f j ) over positive samples greater than its expected value E N (f j ) over negative”

unsupervisedVidClass (unsupervisedVideoCNN)

**exploit visual tracking (thousands of unlabeled YT vids) to learn CNNs in unsupervised way, cos two patches con-**

**nected by a track should have similar visual representation in deep feature space since they probably belong to the same object**

is strong supervision necessary for training CNNs?

Siamese triplet networks = 3 base nets that share the same params

Static imgs not good enough to learn vis reps, but videos are. Is the new thing