DLE Homework 6
Metric Learning

Vojtěch Michal, michavo3

***Abstract*—The assigned homework[1] focuses on retrieval of input images similar to a given query using features generated by the neural net's penultimate layer. All experiments are carried out on the FashionMNIST dataset of 28x28 greyscale images of clothes using a deep convolutional neural network. The retrieval accuracy for several cases is compared – one case when the model is trained for classification (not putting emphasis on embeddings) and two cases when the model is specifically trained to find unambiguous features. Implementation of functions (especially loss functions triplet and smoothAP and statistics AP and mAP) is loop-free and relies only on PyTorch tensor operations.**

I. Image Retrieval

The provided neural net was used in two possible ways in this experiment – either fully evaluate `net(x)` to get 10 class scores or run only `net.features(x)` to get normalized 256-element features useful for image retrieval. The utility function `distances(a, b)` (implemented using the `torch.cdist` for clarity, as the function is simple to understand) computes a matrix of pair-wise squared Euclidean distances between features in `a` and `b`. Given a batch of images and one query image, the algorithm calculates features and their distance from query image features for all elements in the batch. Sorting distances in the ascending order allows selection and retrieval of top 50 images.

A list of images retrieved by the pretrained network trained for classification is shown in Fig. 1. Rows are independent retrievals; the leftmost image is the query, the rest are retrieved images sorted in ascending order by the feature-distance from the query. Retrieved images matching the class of query are marked with green frame, red frame indicates image of a different class. For reproducibility, Fig. 1, 6 and 7 all use query images 993, 859, 298, 553, 672, 971, 27, 231, 306, 706.

[1]The homework assignment is available on https://cw.fel.cvut.cz/wiki/courses/bev033dle/labs/lab6_metric/start

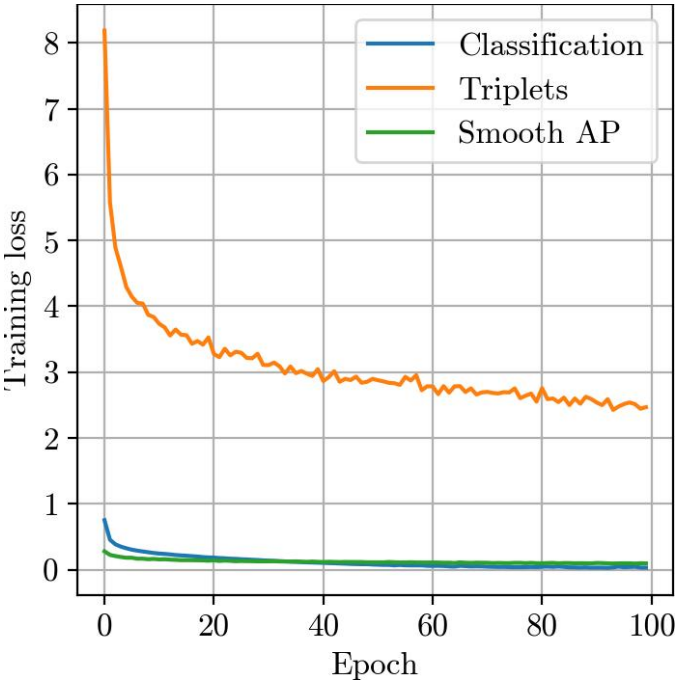

Fig. 2: Mean loss during the training process.

A. Training process

The net was trained with each considered loss function – the cross entropy targeting the classification and two loss functions targeting the retrieval – the triplet loss and the smoothAP loss. The FashionMNIST dataset was split 90:10 into a training and validation set (of sizes 54000 and 6000 samples, respectively); the testing set contains 1000 samples with equal representation of all classes of clothing. Each training process run for 100 epochs using the SGD with several values of hyperparameters.

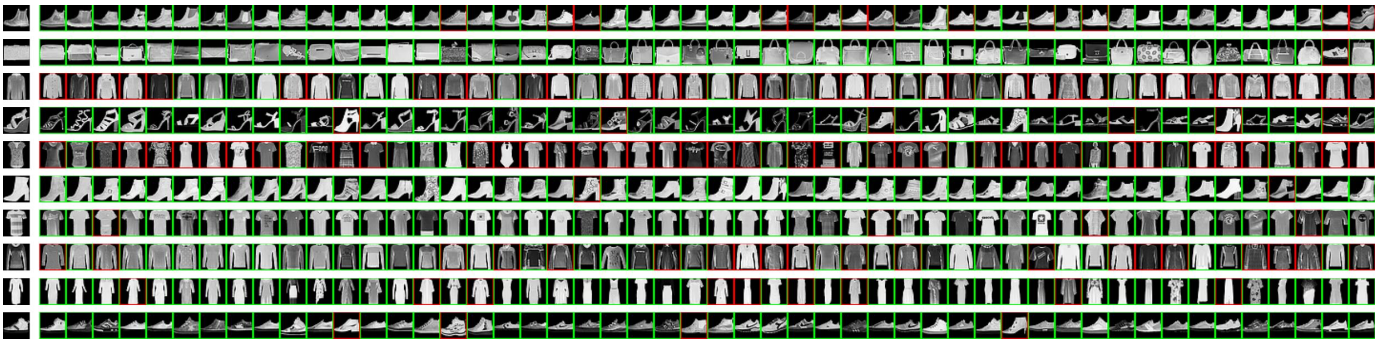

Fig. 1: Closest images retrieved with the net trained for classification.

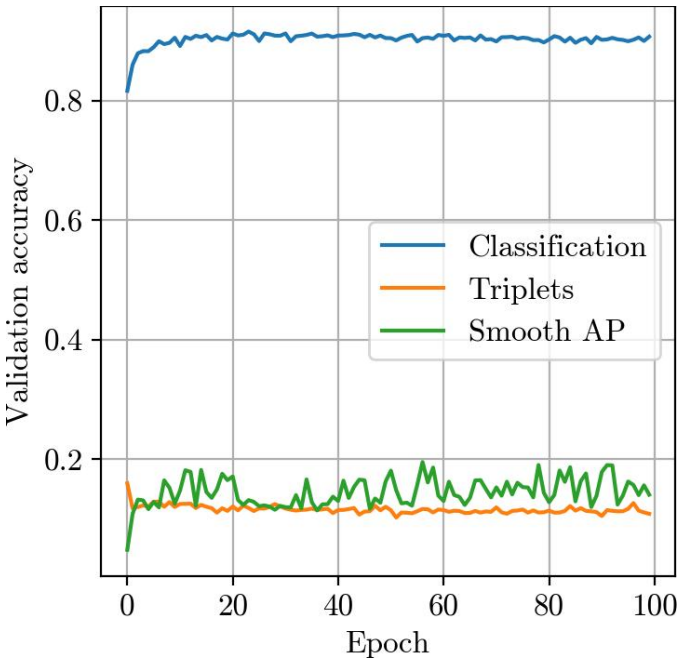

Fig. 3: Validation accuracy during the training process.

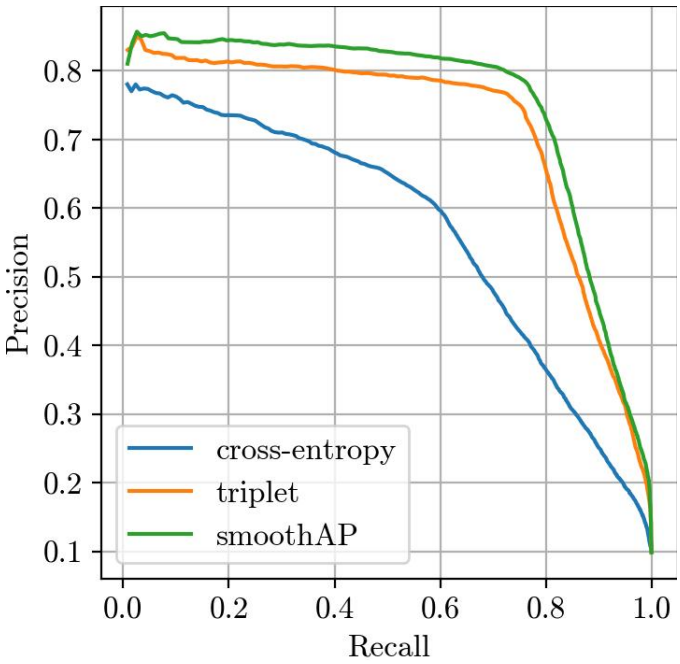

Fig. 4: Precision-recall curve for net training loss functions.

The best results were acquired using momentum 0.9 and learning rate 0.01, which is also the learning rate used in the reference paper [1]. Fig. 2 shows the mean loss recorded during the training processes. It should be noted that values of individual plots should not be compared directly, as each plot corresponds to an entirely different loss function. They are, therefore, mainly shown to verify that the training process improves the net's performance.

An obvious yet initially counter-intuitive fact regarding the validation accuracy shown in Fig. 3 should be mentioned. The validation accuracy – defined as the relative number of samples from the validation set correctly classified by the network – is entirely meaningless when an incompatible loss function is used. When the net is not trained for image classification (that is, it is trained to minimize triplet or smoothAP loss functions), the classification accuracy will indeed be very low. Even worse – the validation accuracy may drop during the learning process, as occurred in the case of triplet loss in Fig. 3. The simple lesson learned is that any classification-related metric is meaningful only as long as the net is trained and used for classification.

B. Evaluation of Embedding

To assess the quality of features produced by the embedding, we use metrics *recall*, *precision* and *Average Precision* (AP, i.e. the area under the recall-precision curve) and average them over 100 random queries from the testing dataset. The mAP achieved using individual loss functions are listed in Table I. All values match the expectation from the assignment (0.58 mAP for cross entropy and roughly 0.8 for both triplet and smoothAP) with possible small variation due to randomness involved in computation of mAP from many samples of AP. The recall-precision curves for all three trained nets is presented in Fig. 4 and the total number of images correctly retrieved by individual nets for individual queries is shown in Fig. 5.

The third query image (sample number 298) was hard to retrieve similar images for, regardless of the loss function. Apart from that query, the number of correctly retrieved images increased by using the triplet loss or the smooth AP loss specifically designed to maximize the mAP statistic. The most noticeable improvement of retrieval performance occurred in the case of 5th query image (sample number 672), where the smoothAP achieved correct retrieval of twice as many images, as can be seen by comparing Fig. 6 and 7.

REFERENCES

[1] Brown, Andrew, and Xie, Weidi, and Kalogeiton, Vicky, and Zisserman, Andrew, *"Smooth-AP: Smoothing the Path Towards Large-Scale Image Retrieval"*, European Conference on Computer Vision (ECCV), 2020

Loss	Achieved mAP
Cross Entropy	0.61
Triplet	0.79
SmoothAP	0.77

TABLE I: Mean Average Precision reached by nets trained using individual loss functions.

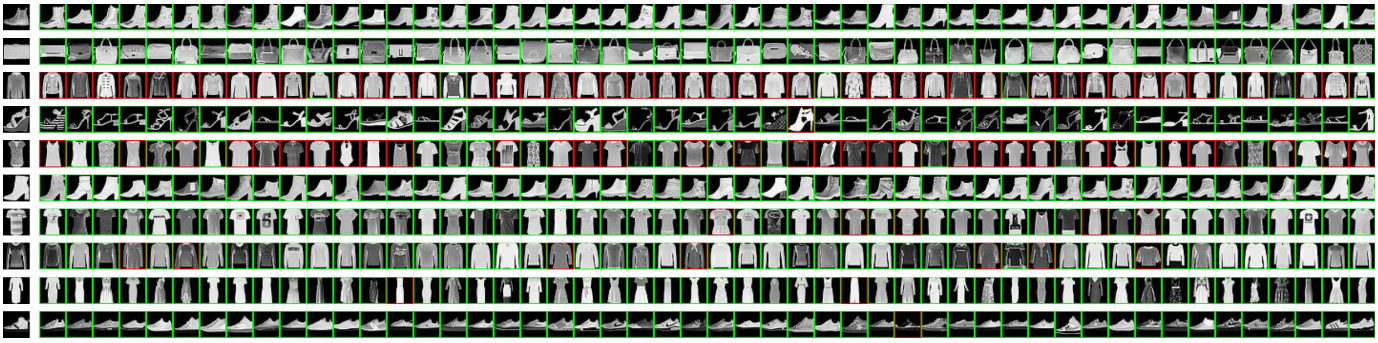

Fig. 6: Closest images retrieved with the net trained using triplet loss.

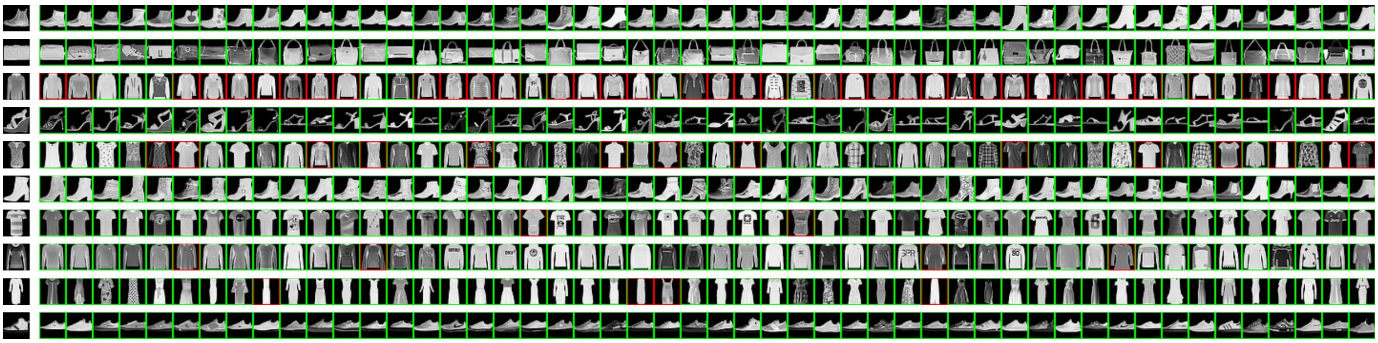

Fig. 7: Closest images retrieved with the net trained using SmoothAP loss.

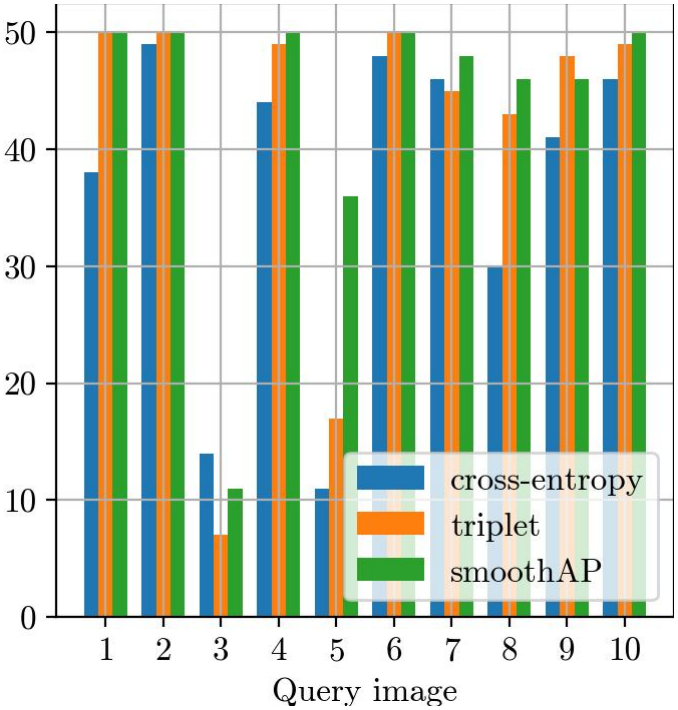

Fig. 5: Number of images correctly retrieved by individual nets for each query.