

# Machine Learning Engineer Nanodegree

## Capstone Proposal

Ann-Kristin Juschka

March 24, 2019

## Proposal

### Domain Background

In this Nanodegree, *Convolutional Neural Networks (CNNs)* are introduced that are most commonly used to analyze visual data. In the deep learning project of the Nanodegree, we train CNNs to detect dogs in images and to classify dog breeds. In general, object detection and classification is a classical task in machine learning.

The aim of this project is to use CNNs to detect and classify brands in images with logos. For this, usually CNNs are trained with the dataset *FlickrLogos-27* [KPT<sup>+</sup>11] and *FlickrLogos-32* [RPLvZ11]; e.g. in [BBMS17, ISGK15]. We are also interested in this task as we later wish to *analyze sentiments* in twitter tweeds with pictures where a given company's logo is detected. Due to time and size limits, we focus in this project on *logo detection and classification*.

### Problem Statement

While most logo datasets like *FlickrLogos-27* [KPT<sup>+</sup>11] and *FlickrLogos-32* [RPLvZ11] contain raw original logo graphics, we want to train our CNN on the in-the-wild logo dataset *Logos in the Wild* [THMB18] whose images contain logos as natural part. As this dataset includes in total 11,054 images with 32,850 annotated logo bounding boxes of 871 brands, it should be possible to train a CNN that achieves a high accuracy or mean average precision (map). This is a challenging task as the regions containing logos are small.

### Datasets and Inputs

To train our model, we want to use the recent logo dataset *Logos in the Wild* [THMB18]. As mentioned above, this the latest version of this dataset (v2.0) contains 11,054 images with 32,850 annotated logo bounding boxes of 871 brands and it is collected by performing Google image searches with well-known brand and company names directly

or in combination with a predefined set of search terms like ‘advertisement’, ‘building’, ‘poster’ or ‘store’. The logo annotations are in Pascal-VOC style.

Moreover, this dataset has 4 to 608 images per searched brand, 238 of 871 brands occur at least 10 times, and there are up to 118 logos in one image. Unfortunately, the dataset provides only the links to the images, and some of these images already disappeared. As we want to later detect logos in arbitrary pictures from twitter tweeds, this large in-the-wild logo dataset still fits best to our goal.

## Solution Statement

The main goal of this project is to use CNNs to *classify the brand and company logos* from the Logos in the Wild dataset with high accuracy and mean average precision (map). For this, we first train an own CNN architecture as done in the dog breeds project of the Nanodegree. Furthermore, to improve loss and accuracy we make use of the standard CNN architectures *VGG19*, *Resnet50*, *InceptionV3*, or *Xception* that are already trained on the *ImageNet* database [ker]. Moreover, as the dataset comes with annotations in Pascal-VOC style, if time permits we will also train a *Faster Region-based Convolutional Neural Network (Faster R-CNN)* [RHGS15] for two stages: First, an *Region Proposal Network (RPN)* for *logo detection*, and second, a classifier like VGG19 for logo classification of the candidate regions. This Faster R-CNN will be evaluated using the *mean average precision (map)* metric.

## Benchmark Model

The recent *Logos in the Wild* dataset has not been studied much yet. When introduced in [THMB18], the focus is put on open set logo retrieval where only one sample image of a logo is available.

Instead we want to focus on a closed world assumption where we train and test on the Logos in the Wild dataset which has multiple images per brand. Therefore, we can only compare our results with the ones of other models that were trained and tested on the popular closed dataset *FlickrLogos-32* [RPLvZ11]. As cited in [THMB18], the mean average precision (map) in state-of-the-art results is 0.811 achieved by Faster-RCNN [SZG16] where the training set is expanded with synthetically generated training images, and 0.842 using Fast-M [BLF<sup>+</sup>16] that is a multi-scale Fast R-CNN based-approach.

## Evaluation Metrics

For purely logo classification, we seek to achieve a high accuracy. *Accuracy* is simply the number of correct predictions divided by the total number of predictions.

For logo detection, we need a different metric: mean average precision (map) that is the mean over all classes, of the interpolated *average precision* [EVGW<sup>+</sup>10] for each class. Recall that

$$\text{precision} = \frac{\text{true positives}}{\text{true positives} + \text{false positives}}, \quad \text{recall} = \frac{\text{true positives}}{\text{true positives} + \text{false negatives}}.$$

Considering the precision-recall curve for a given threshold, and the interpolated precision  $\text{precision}_{\text{interpolated}}(\text{recall}_i) = \max_{\tilde{r}, \tilde{r} \geq \text{recall}_i} \text{precision}(\tilde{r})$  of the 11 values  $\text{recall}_i = \{0, 0.1, 0.2, \dots, 0.9, 1.0\}$ . Then

$$\text{average precision} = \frac{1}{11} \sum_1^{11} \text{precision}_{\text{interpolated}}(\text{recall}_i) \quad [\text{EVGW}^+10, \S 4.2].$$

## Project Design

At first, we will collect the images through the provided urls in the Logos in Wild dataset. Probably we augment the training data to have more images per brand.

As mentioned before, we start by training an own CNN architecture to classify the logos as done in the dog breeds project of the Nanodegree. For this, 40% of the dataset is forms the test set, and the remaining data is split into 80% training set and 20% validation set. Using Keras preprocessing, each image is converted into a 4D tensor with shape  $(1, 224, 224, 3)$ . This CNN may consist of a convolutionary layer, a max-pooling layer, another convolutionary layer, a max-pooling layer, a global-average pooling layer and a final fully connected layer. As optimizer we choose "RMSprop", as loss function "categorical cross entropy" and as metric "accuracy". Using Keras ModelCheckpoint, we save the model with the best validation loss.

As a next step we train a popular CNN like VGG19 whose weights are pre-trained on ImageNet. We proceed similarly as in the first setting.

Having tried CNNs for logo classification, we finally proceed to Faster R-CNNs for logo detection and classification. We first train a *Region Proposal Network (RPN)* that proposes regions with logos in images. The predicted region proposals are then reshaped using a *Region of Interest (RoI)* pooling layer. This layer is next used to classify the image within the proposed region and predict the offset values for the bounding boxes. For the latter task, we again train a CNN like VGG19 that is pre-trained on ImageNet. As explained above, here we use mean average precision as metric.

## References

- [BBMS17] Simone Bianco, Marco Buzzelli, Davide Mazzini, and Raimondo Schettini. Deep learning for logo recognition. *Neurocomputing*, 245:23 – 30, 2017.
- [BLF<sup>+</sup>16] Yu Bao, Haojie Li, Xin Fan, Risheng Liu, and Qi Jia. Region-based cnn for logo detection. In *Proceedings of the International Conference on Internet Multimedia Computing and Service*, ICIMCS'16, pages 319–322, New York, NY, USA, 2016. ACM.
- [EVGW<sup>+</sup>10] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010.

- [ISGK15] Forrest N. Iandola, Anting Shen, Peter Gao, and Kurt Keutzer. Deeplogo: Hitting logo recognition with the deep neural network hammer. *CoRR*, abs/1510.02131, 2015.
- [ker] Keras Documentation. <https://keras.io/applications/>. Online; accessed 2019-03-17.
- [KPT<sup>+</sup>11] Y. Kalantidis, LG. Pueyo, M. Trevisiol, R. van Zwol, and Y. Avrithis. Scalable triangulation-based logo recognition. In *in Proceedings of ACM International Conference on Multimedia Retrieval (ICMR 2011)*, Trento, Italy, April 2011.
- [RHGS15] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster R-CNN: towards real-time object detection with region proposal networks. *CoRR*, abs/1506.01497, 2015.
- [RPLvZ11] Stefan Romberg, Lluís Garcia Pueyo, Rainer Lienhart, and Roelof van Zwol. Scalable logo recognition in real-world images. In *Proceedings of the 1st ACM International Conference on Multimedia Retrieval, ICMR '11*, pages 25:1–25:8, New York, NY, USA, 2011. ACM.
- [SZG16] Hang Su, Xiatian Zhu, and Shaogang Gong. Deep learning logo detection with data expansion by synthesising context. *CoRR*, abs/1612.09322, 2016.
- [THMB18] Andras Tüzkö, Christian Herrmann, Daniel Manger, and Jürgen Beyerer. Open Set Logo Detection and Retrieval. In *Proceedings of the 13th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications: VISAPP*, 2018.