Karlsruhe Institute of Technology

Fraunhofer IOSB

Logo Retrieval in Mass Data Using Deep Learning

MASTER THESIS

KIT - KARLSRUHE INSTITUTE OF TECHNOLOGY
FRAUNHOFER IOSB - FRAUNHOFER INSTITUTE OF OPTRONICS,
SYSTEM TECHNOLOGIES AND IMAGE EXPLOITATION

Andras Tüzkö

June 14, 2017

Advisors:
Prof. Dr.-Ing. habil. Jürgen Beyerer
Dipl.-Inform. Christian Herrmann
Dipl.-Inform. Daniel Manger

Interaktive Echtzeitsysteme

Statement of authorship

I hereby declare that I have produced this work by myself except the utilities known to the supervisor, that I have labeled all used utilities completely and detailed and that I have labeled all material that has been taken with or without modification from the work of others.

Karlsruhe, June 14, 2017 Andras Tüzkö

Abstract

Abstract

In this work, different deep learning based solutions will be introduced to retrieve open-set logos by searching for correspondences in the query image set. The search context are very challenging sport videos, with logos being often blurred and having a low resolution. The problem is made further difficult by the small number of datasets, having a low number of ground truth examples. For this purpose, a self annotated dataset will be introduced from the target domain. Different experiments are performed, to increase the size of the training data with synthetic data. The introduced system generates region proposals for possible object locations with Faster R-CNN, which is a state-of-the-art object detection framework. The final system surpasses the performance of the detailed baseline method by a large margin, which is the standard for closed-set logo retrieval.

Kurzfassung

In dieser Arbeit werden verschiedene Deep Learning basierte Lösungen vorgestellt, um unbekannte Logos in Videos mit einer Menge von Anfragebildern zu suchen. Die Datenquellen sind herausfordernde Sportvideos mit niedrig aufgelösten Logobildern, die oft auch verschwommen sind. Das Problem wird weiter erschwert, wenn man die kleine Größe der öffentlich verfügbaren Datensätze mitberücksichtigt. Deshalb wird ein selbst annotierter Datensatz vorgestellt, welcher aus mehreren Sportvideos extrahiert wurde. Um die Größe des Trainingsdatensatzes zu vergrößern, werden verschiedene Experimente bezüglich synthetischer Daten durchgeführt. Das vorgestellte System verwendet Faster R-CNN um Vorschläge von Regionen für mögliche Stellen von Objekten zu generieren. Faster R-CNN ist ein hochmoderner Objektdetektor und-klassifikator und wird standardmäßig für Closed-Set Logo Suche in Bildern eingesetzt. Das finale System übertrifft die Performance der vorgestellten Baselinemethode mit einem großen Abstand.

Contents

1

Introduction

Advertising with static logos is one of the most important marketing methods. A very effective way to reach a lot of people with these static logos is to sponsor sports teams or to buy advertising spaces in sports events broadcasted on the TV. However, the prices of these surfaces mean huge expenses for the advertiser. This is the reason why the need for logo appearance statistics of sport videos arises. In particular there is a desire for quantitative measurement of the proportional size of the logo to the screen and of the time one particular logo is visible on the screen. This data is then used to judge the cost efficiency for the specific logo placement, i.e. to be able to decide on which sports event to advertise, with which size of logo and where to place it.

In this work, a system for logo retrieval with proposal based object detection and classification will be presented. The system consists of a logo detector and a classifier used for feature extraction. The logo detector is a Faster Region-Based Convolutional Neural Network [Ren15] trained to recognize logos on images. For feature extraction from the proposed region, there will be the performance of several classifier neural network tested. A similarity score will be calculated to decide the identity of the logo to be searched and the proposed region. To recognize logos in videos, a video will be cut into frames and then the system will be run on every image. Figure 1.1 shows the structure of the system. The challenge of this task is manifold. The first problem is that the logos in these videos are

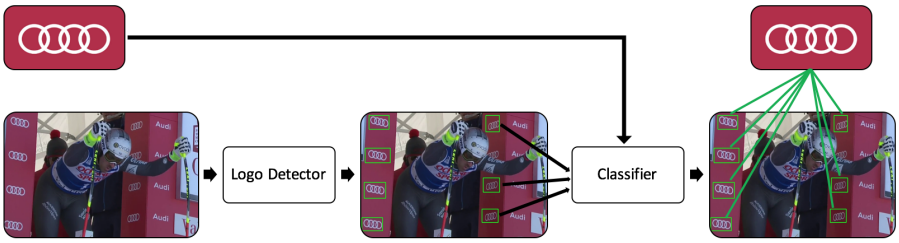

Figure 1.1: The outline of the proposed system

far from being perfectly clear. They can be partially occluded, blurred - if the camera moves fast, perspectively transformed, rotated and can have various coloring, suiting well to the design of the shirt or the arena. In addition, there is a problem with the ambient illumination variation just as for other computer vision tasks. The second challenge is due to the legal prohibition of copying symbols. This already results in a huge appearance variety, which is further grown by the diversity of logos within a company. This makes the detection of logos very complex. In addition, a lot of companies use wordmarks i.e. a logo only with its name. For a system, not prepared for text recognition, it is a harder task to recognize complete words than only a letter or a simple graphic mark, symbol. These problems make the task of logo retrieval to an open-set recognition problem. Furthermore, in classification tasks, the objects to be classified have usually a 3D shape in the reality, whereas

logos have only a planar surface. It means, that it does not yield additional information, if a logo is photographed from multiple different angles, unlike in case of other objects. Unfortunately, there are only a few publicly available small datasets with bounding box annotated logos. The majority of the images are adjusted to ensure a good visibility of the logos, unlike on the frames of the sport videos. Figure 1.2 introduces some examples of the challenges. To master these challenges, the available

Figure 1.2: Examples for challenging logos, where the instances of each column belong to the same class

solutions and methods improved a lot recently. In the decade before, hand-crafted feature extraction was prevalent in computer vision tasks. It needed an expert to create a system and it yielded often only mediocre results. Deep learning methods for computer vision problems have been dominant since the success of Convolutional Neural Networks in 2012 [Kri12]. Compared to earlier systems (e.g. Scale and Translation Invariant Features [Low04], Histogram of Oriented Gradients [Dal05] features), the great improvement of deep learning methods is the capability of learning how to extract features automatically. The enhancements are encouraged by continuous research. The development of deep nets is mainly powered by the annually organized ImageNet classification challenge [Rus15]. Since the aim of this contest is to classify an object, which fills out the majority of an image, the location of the particular object is irrelevant. To be able to classify and recognize objects, which have a much smaller size relative to the size of the whole image, region-based classification can be utilized.

The rest of this thesis is organized as follows. Chapter 2 reviews the related work of image retrieval, object detection and logo retrieval. In chapter 3 the proposal based object detection with Convolutional Neural Networks will be introduced. Chapter 4 describes the logo retrieval system. Chapter 5 presents some experiments to extend the size of the dataset, and evaluates different methods on a standard dataset. Afterwards, chapter 6 includes evaluation and comparison of the system with another logo retrieval methods. Finally, the last chapter concludes the work and gives prospects on future work.

2

Related Work

This chapter gives an outline on the recent research related to this work. The included topics are deep learning, object detection, image retrieval, open-set classification and logo retrieval. Deep neural networks are motivated by the memorization procedure of the human brain. Rosenblatt invented perceptrons [Ros58], which serve as the basis of today's Fully Connected layers (FC) in deep learning. LeCun applied [LeC89] Convolutional Neural Networks (CNN) for handwritten digits recognition in 1989, which are inspired by the visual cortex of the nervous system. Since then, a lot of research was made to improve and extend the application scope of deep learning in computer vision.

2.1 Object Detection

Earlier systems utilized hand-crafted features to detect objects on images and recognize them. Lowe et al., used Scale and Translation Invariant Features (SIFT) [Low04] around keypoints, detected with e.g. Harris corner detector [Har88]. Viola and Jones utilized [Vio04] Haar-like features and a cascade of weak classifiers (Adaptive Boosting [Sch99]) to detect faces extremely fast. Nowadays, deep learning methods surpass the traditional methods by a wide margin. OverFeat framework [Ser13] uses sliding windows on multiple scales of the image, and combines the features to detect objects and to classify them. You Only Look Once (YOLO) [Red15] introduces an end-to-end network for detecting and classifying objects by using bounding box regressors at the first time for localization. It splits the input image into a square grid, where every cell predicts several bounding boxes with probability scores and classification labels. The Single Shot MultiBox Detector (SSD) [Liu15] utilizes convolutional features from multiple layers and concatenates them to detect objects in real time. Faster Region-Based Convolutional Neural Network (R-CNN) [Ren15] will be detailed in chapter 3. Region-based Fully Convolutional Network (R-FCN) [Dai16] is the improvement of Faster R-CNN in terms of inference time by having a network end-to-end fully convolutional. Mask R-CNN [He17] extends the functionality of Faster R-CNN by extending the network with a classification mask, which allows end-to-end object detection and semantic segmentation with a little overhead.

2.2 Image Retrieval

Many techniques exist outside the scope of deep learning for image retrieval from videos. SIFT features [Low04] with bag-of-visual-words were used to get translation invariant descriptors around keypoints by Zisserman [Siv03]. The visual words were then used to retrieve objects in videos instantaneously like searching on Google. Histogram of Oriented Gradients (HOG) [Dal05] descriptors are gradient orientation histograms, extracted blockwise as features. Hu et al. [Hu13] used an extended version of HOG descriptors to retrieve images based on sketches from a database. Nowadays,

deep learning approaches are prevalent in the context of image retrieval. In [Yan16], CNN and SIFT features were compared and fused to retrieve images. Babenko et al. [Bab14] utilized the output of the middle fully connected layer to retrieve images taken in same or similar scenes. This approach is used also in this work, however Babenko used the complete image as input, not low resolution Region of Interests (RoIs). Gordo et al. [Gor16] extracted local features from different regions of an image, which are selected by a region proposal system, and combined them to a global feature to retrieve images with similar scenes. In [Rad16] Maximum Activations of Convolutions (MAC) was computed from the output of a Fully Convolutional Network to represent images. MAC is a max pooling from the two dimensions of each channels, then the max value of every channel is used as a feature.

2.3 Open Set Classification

Bendale et al. argue [Ben15] with the fact that the majority of proposed neural networks utilize softmax layer to get classification probabilities. Thus, these networks are inherently trained for a closed-set classification world. These models can easily be fooled with abstract images, humans easily reject from any of the trained classes, but with the network predicting a class label with high probability. For this purpose, they propose OpenMax, which can be used to replace the softmax layer. OpenMax is able to classify such fooling and adversarial images as unknown.

2.4 Logo Retrieval

Romberg et at. [Rom11a] published FlickrLogos-32 dataset, which became the standard evaluation dataset of logo retrieval systems. Furthermore, Romberg et al. [Rom13] generated synthetic data to increase the training dataset size, and combined local features from logo images into an aggregated feature. Pandey et al. [Pan14] used SIFT features and bag-of visual words to retrieve logos from natural images, where the logo filled the complete image. In [Hoi15] a large logo dataset of 100 brands with about 130,000 instances was introduced, which is unfortunately still not publicly available. In [Egg15], synthetic logo dataset was utilized and a neural network was trained with unlabeled data by bootstrapping. Iandola et al. [Ian15] used Fast R-CNN for the first time to retrieve logos from images. Furthermore, R-CNN, Fast R-CNN and Faster R-CNN were used in [Bao16], [Oli16], [Qi17]. In [Su16] synthetic logo data was used to extend a train dataset with very scarce size. In 2017 Bianco et al. introduced Logos-32Plus [Bia17], which is currently the largest publicly available dataset with 12,312 RoIs altogether.

3

Proposal Based Object Detection and Classification

In this section, the theoretical overview of the proposed logo retrieval system will be presented. First of all, the advanced Convolutional Neural Networks are detailed in section 3.1. The explained networks are later evaluated in chapter 5. Fully convolutional networks will be introduced in section 3.2. Section 3.3 explains region proposal systems for generating candidate object locations on an image. Afterwards, section 3.4 describes Region-Based Convolutional Neural Networks for object detection and classification. The improvement of this method, the Fast Fegion-Based Convolutional Neural Networks will be detailed in the section 3.4.2. Following this, the further development of this framework, the Faster Region-Based Convolutional Neural Networks will be reviewed in the section 3.4.3.

3.1 Modern Convolutional Neural Networks

Convolutional Neural Networks are inspired by the visual cortex of humans, introduced by LeCun et al. [LeC89], for zip code recognition. Image classification challenges are dominated by Convolutional Neural Networks since the success of AlexNet architecture [Kri12] in ImageNet Large Scale Visual Recognition Challenge (ILSVRC) contest [Rus15]. The ImageNet dataset contains 15 million images, in the challenge only a subset of that is used. The task in this challenge is to classify 100,000 images of 1,000 categories, after training with 1.2 million annotated images.

AlexNet was the first neural network after the conquest of support vector machines, achieving impressive performance, and won the ImageNet challenge in 2012. It consists of five convolutional layers, each followed by a max-pooling, which counted for a very deep network at that time. In this network, Rectified Linear Units (ReLU) layers are utilized as activation function. This results in much shorter training times than *tanh*, which is susceptible to saturate. It uses dropout layer [Hin12] as a regularization method against overfitting, which allows to train much deeper networks. It simulates the presence of more neural networks, each inferring a different convolutional map from the input image. The network ends with three fully connected layers, having 1,000 outputs for the 1,000 classes to be classified in the last layer.

In the following year, in 2013, Zeiler and Fergus proposed ZFNet [Zei13], which takes AlexNet as basis, but outperforms it slightly. They implemented feature activation visualization, which helped to reveal several problems of AlexNet. From the lessons learned, they applied smaller filter sizes (7x7, instead of 11x11) and the stride is also lowered from four to two in the first convolutional layer.

Szegedy et al. introduced GoogLeNet [Sze14], which is a much deeper network and won the classification challenge in 2014. This network utilizes "Inception" blocks, which embody three

convolutional layers with different filter sizes, being responsible for the features of diverse sizes in the same input convolutional map. In addition, a block comprises also a max pooling branch, to obtain small translation invariance. The output of all contained layers is concatenated together, and serves as the output of the block. In the same year, VGG network was proposed too [Sim14], which won the first place in the localization part of the challenge. This network follows the general architecture of AlexNet by utilizing also five max-pooling layers, and having three fully connected layers at the end. However, the number of convolutional layers are increased to 13, whereas the filter sizes are reduced further to 3x3. Convolutional strides are fixed to one, to preserve every information and avoid losing performance.

A medium sized network, called `VGG_CNN_M`, was proposed in [Cha14]. It is very similar to the ZFNet, but has reduced stride and receptive field in the first convolutional layer. For the majority of the experiments in chapter 5, a modified version of this network is used, called `VGG_CNN_M_1024` [Cha14], having only 1,024 outputs instead of 4,096 in the second last fully connected layer.

Residual Networks (ResNet) [He15] utilize very deep network architecture of 50-151 layers, and won the contest in 2015. It adopts skip connections, which is a direct connection from the output of a lower layer (the one lying closer to the input image). These connections address the problem of degradation, which arises when very deep networks are involved, causing performance dropping. It eliminates the need of learning to map the identity of earlier layer's output. It applies Batch Normalization (BN) layer for regularization and input normalization, proposed in [Iof15]. This is an intermediate normalization layer, eliminating the need of mean subtraction and division with the variance of the input images, buy learning shift and scaling parameters applied on the input map. This means, that the network can learn these parameters specific for every layer, and the normalization can be build into the network in an end to end manner. Since the introduction of BN, it became a widely applied normalization technique in a lot of new proposed networks, but improves also older networks e.g. GoogLeNet has also a variant utilizing BN.

In 2016, ResNets were further improved by [Xie16], called ResNext - Residual networks with next dimension. This network won the second place on the challenge, where the first place is won by Team Trimps-Soushen [ILS16] with an ensemble of classifiers. ResNext splits a ResNet unit into a multi-branch Convolutional Neural Network, achieving better performance than earlier networks.

DenseNet [Hua16] has also a ResNet-like architecture, but it rather connects the output of a layer with every subsequent layer's input, This results in a much denser network than a conventional feed-forward network, having $L \cdot (L-1)/2$ connections, instead of L, where L is the number of layers or the number of "dense units" respectively. This is accomplished by concatenating instead of summarizing the output of a unit with the previous output.

3.2 Fully Convolutional Neural Networks

A neural network is fully convolutional, if it does not contain any fully connected layers. Firstly Matan et al. [Mat92] used FCNs for recognizing strings of digits. Long et al. [Lon14] proposed how to transform a deep neural network with fully connected classifier layers at the end, to a Fully Convolutional Network. For this purpose, the fully connected layers at the end of the network are converted to convolutional layers. Particularly, this is achieved by considering a convolutional layer as a fully connected layer applied in sliding window fashion on the input map. For this purpose, the fully connected layers are replaced with convolutional layers, having a filter size equal to the size of the input feature map in the original network. As a result of this, if the new network is fed with an

input image, larger than the original input size of the network, the convolutional layer will process the feature map, and the output is not a vector, but a grid.

Since the number of weights of a neuron in a fully connected layer is defined by the shape of the data of the layer, the trained network, containing fully connected layers, can process only a fix-sized input image. As a Fully Convolutional Network does not have fully connected layers anymore, it has the advantage of being able to train and test with images of arbitrary sizes.

The outputs of such a network are two dimensional feature maps, which can be used as heatmaps per class, whereas the higher activations mean the presence of that specific class. These convolutional maps can also be used directly for semantic segmentation, where each pixel of an image should be classified. Nowadays, fully convolutional networks are an essential part of state-of-the-art object detectors, yielding better performance, image size agnosticism, as well as shorter training and inference times as shown in section 3.4.

3.3 Region Proposal Systems

To recognize different objects on an image like logos, small regions should be considered. The easiest way to search for these locations is the exhaustive sliding window search, applied on multiple scales. However, this induces a lot of computational costs. In order to reduce the number of regions, thus the computational burden, region proposal systems can be utilized. Region proposals are possible object locations on an image.

Earlier computer vision solutions used external proposal systems. This means that the proposals of every image should be pre-calculated before training or inference. One of the most popular region proposal methods is Selective Search [Uij13]. It merges neighbor regions according to a similarity score in a bottom-up fashion. It processes an image under 2s on the CPU, which excludes the possibility of real time applications. Edge Boxes [Zit14] are efficiently calculating the number of contours in a box, and ranking them according to that almost in real time. Recently, as section 3.4.3 introduces, the proposal system is already part of the neural network.

3.4 Region-Based Convolutional Neural Networks

This section gives a brief overview about how Faster Region-Based Convolutional Neural Networks evolved.

3.4.1 Regions with Convolutional Neural Network Features

Although this network is already historical, it is worth mentioning it, because it helps to understand the improvements of the later systems. Region-Based Convolutional Neural Networks [Gir13] consist of four separate systems. Firstly, region proposals are generated externally, with Selective Search. Altogether 2,000 object positions are considered. Secondly, each region of the possible object locations is warped to a size of 227x227, and then the feature vector of every single region is extracted with a CNN. The network [Kri12] is pretrained on the ImageNet dataset [Den09], and then fine-tuned on the final classes. The network is run on every region proposal bounding box, to extract vectors with a fixed-size. These vectors will be written to the disk. Thirdly, a set of class-specific linear Support Vector Machines (SVM) [Cor95] is used to classify the specific region. At last, bounding box regression is run, to reduce the mislocalization of the object. This happens outside of the network. Figure 3.1 introduces the functioning of the network.

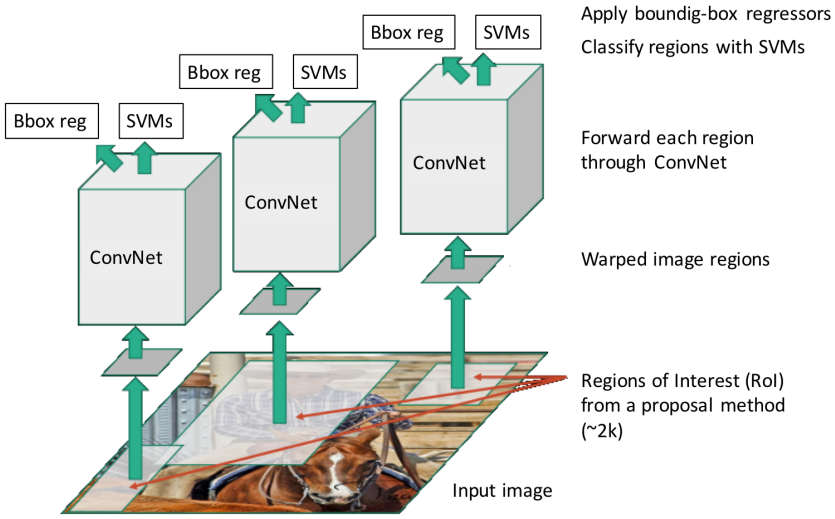

Figure 3.1: R-CNN takes external region proposals, warps the region to a uniform shape, and extracts the features separately. The extracted features are then used to classify the region, and calculate bounding box regression externally. Source: [Gir16a]

3.4.2 Fast Region-Based Convolutional Neural Network

Fast Region-Based CNNs [Gir16b] are aimed to improve the classification accuracy and feature vector extraction speed of the interest regions, generated also with Selective Search. An intermediate convolutional feature map is extracted from the whole input image with a Fully Convolutional Neural Network, also called base network in [Ser13]. The output is a downscaled feature map, which is fed to the RoI pooling layer. This layer crops regions from the map according to the appropriate downscaled region proposals, and executes a modified version of max pooling on each region, which results in a convolutional map with a fixed-shape, regardless the size of the region.

After the pooling, fully connected layers are used to calculate the final class probabilities and bounding box regressions for each region. The outputs of the bounding box regression are class specific small position and size adjustments, needed to refine the rough object locations.

The proposed network is trained for $K+1$ classes, where K is the number of object classes, and the background is also modelled as a separate class. During training, the positive examples are chosen, regarding the Intersection over Union (IoU) value to the ground truth. This value is widely used for measuring the overlapping between regions, regardless of the actual size of the regions. The calculation between two regions, R_1 and R_2 is as follows:

$$IoU = \frac{area(R_1 \cap R_2)}{area(R_1 \cup R_2)} \tag{3.1}$$

For positive training examples, there are those regions applied, which have an IoU with the ground truth at least 0.5. For the background class, the examples with IoU [0.1,0.5) are used.

The network is trained with a multi-task loss function for classification and bounding box regression, defined as:

$$L(\mathbf{p}, u, \mathbf{T}^u, \mathbf{v}) = L_{cls}(\mathbf{p}, u) + \lambda[u \geq 1] L_{loc}(\mathbf{T}^u, \mathbf{v}) \tag{3.2}$$

where $\mathbf{p}$ is the computed class probability vector, u is the ground truth class, $\mathbf{T}^u$ is the predicted bounding box offset vector for every class, and $\mathbf{v}$ is the ground truth bounding box position and size. The probabilities are calculated with softmax as usual, so the log loss is used for the classification error: $L_{cls}(\mathbf{p},u) = -\log \mathbf{p}_u$. Since the bounding box regressor cannot be trained with background images, its error is not added to the complete loss. This is achieved by setting the label of the background class to zero, and using the term $[u \geq 1]$, which is 1 if $u \geq 1$, and 0 otherwise. For regression loss, a smooth version of L1 loss is used, which is defined as follows:

$$L_{loc}(\mathbf{T}^u, \mathbf{v}) = \sum_{i \in (x,y,w,h)} smooth_{L_1}(\mathbf{t}_i^u, \mathbf{v}_i) \tag{3.3}$$

$$smooth_{L_1}(x) = \begin{cases} 0.5x^2 & \text{if } |x| < 1 \\ |x| - 0.5 & \text{otherwise} \end{cases} \tag{3.4}$$

The improvements of this method compared to the previous Region-Based CNN introduced in section 3.4 are as follows:

- **Joint feature extraction:** much shorter training and inference time is achieved by the lower computational redundancy of running convolutional layers on the whole image only once rather than for every proposed region.
- **One network:** the feature extraction and the classification happens in the same network. This has more advantages:
  - This results again in faster test and training times, due to the unnecessity of writing the extracted feature vectors to disk, which incidentally could require e.g. hundreds of gigabytes of storage [Gir16b] for the VOC07 trainval set [Eve07].
  - As the backpropagation is implemented through the RoI pooling layer, the whole network, together with the convolutional layers, can be trained jointly, against earlier implementations like R-CNN [Gir13] or spatial pyramid pooling networks (SPPnet) [He14].
- **Minibatch from a few images:** Faster training speed is achieved by collecting a minibatch only from two images rather than every region from different images. This method has been proved to converge within similar times despite the high correlated regions.

Figure 3.2 shows the main components of the network.

3.4.3 Faster Region-Based Convolutional Neural Network

A significant disadvantage of the Fast R-CNN method is, that the region proposals are generated externally. Girshick et al. [Ren15] introduces Faster R-CNN, which generates the interest regions within the neural network nearly cost-free (10ms pro image). This system consists of the combination of a region proposal system and a Fast R-CNN object detector.

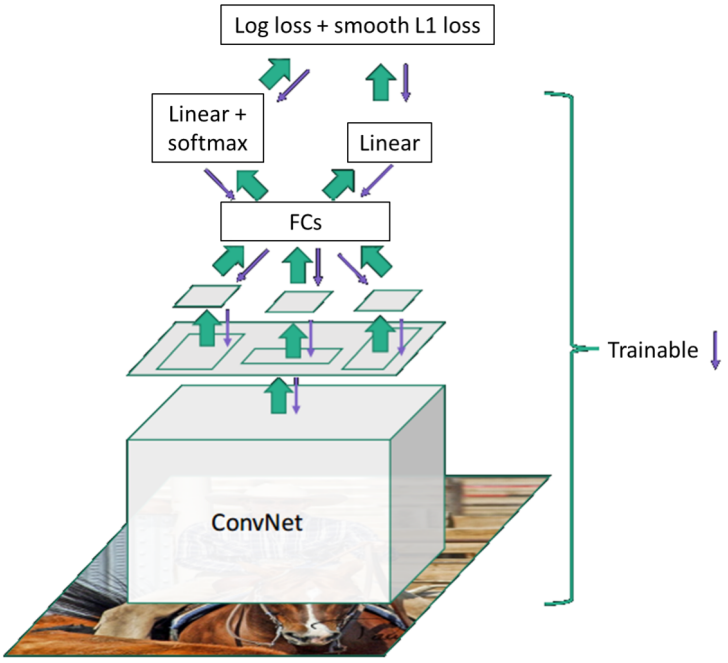

Figure 3.2: Fast R-CNN uses external proposals, then infers the complete image with a fully convolutional network. Afterwards, the proposals is used to crop regions from the feature map with RoI pooling. The cropped region is then classified and the region coordinates are adjusted with a fully connected network. Source: [Gir16a]

Region Proposal Network

The convolutional feature map, extracted by the base network, is processed by the RoI pooling layer. Additionally a thin fully convolutional network, the Region Proposal Network (RPN) is also utilized on the convolutional maps, to generate the proposals. Since it is a FCN, it scans the input in a sliding window fashion. However, it works much faster than earlier systems, applying this technique, because this sub-network has a very low inference time and it is utilized on the compressed convolutional maps, not on the complete input image. Reference boxes, called "anchors" are generated at every position of the convolutional feature map in different scales and different aspect ratios. This ensures the scale invariance of the objects. A convolutional layer with 3x3 kernel extracts a fixed-size vector from every window of the convolutional map, where in each window nine anchors are considered. As the fully convolutional network iterates through the convolutional map in a sliding window fashion, translation invariance is granted. An objectness score and bounding box offset is then calculated for every anchor, by different classifiers and regressors, specialized in a specific scale and aspect ratio. Figure 3.3 demonstrates how an image is processed by the network.

Training

As the RPN is a class agnostic object detector, it should detect all types of objects the network is trained on. For this purpose, during training of the RPN, the class information can be discarded. Positive examples are collected from the proposals with an IoU higher than 0.7 with any ground truth. Negative label is assigned for regions, which have an IoU, lower than 0.3.

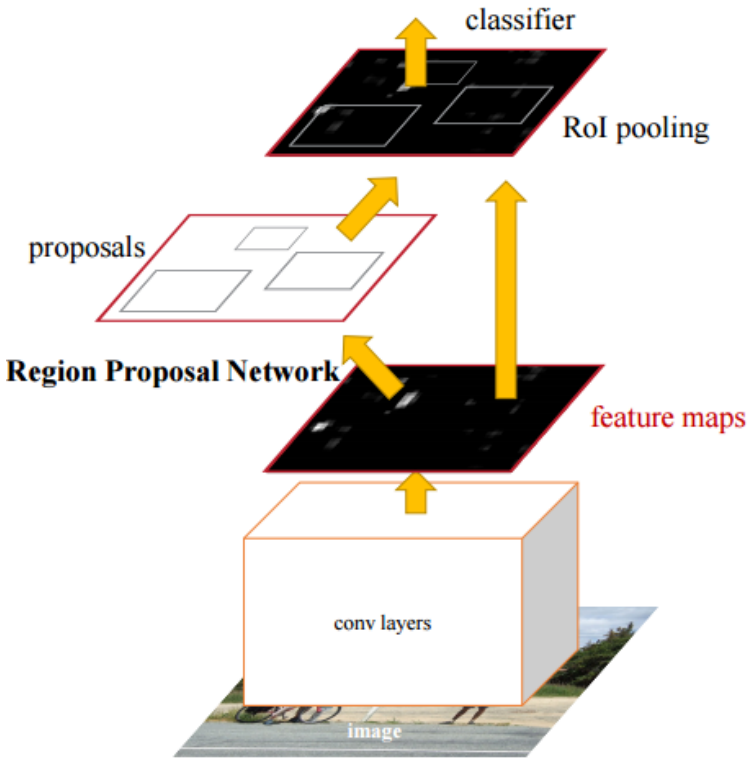

Figure 3.3: Faster Region-Based Convolutional Neural Network consists of a Fast R-CNN and a Region Proposal Network. Fast R-CNN is responsible for the feature map extraction from the whole image, and classify regions from that. The RPN is an in-network implemented proposal system, for generating candidate object locations in a fast way. Source: [Ren15]

Since the objective of the RPN is the same as for Fast R-CNN, namely to classify regions and regress bounding box coordinates, the loss functions for training the RPN are the same multi-task losses, which are used for training a Fast R-CNN, detailed in section 3.4.2.

Earlier versions of the system suggested to train the complete system in an alternating way. First the RPN will be trained with a CNN, pretrained on e.g. the ImageNet dataset [Den09]. After that, the trained RPN will be used to propose regions, and train the Fast R-CNN part with them. The CNN of the fine-tuned Fast R-CNN is then used to train the RPN again, and this process is repeated iteratively. Later, it has been proven, that training the whole network of the Faster R-CNN can be performed in an end to end manner, with marginal performance drop. This means, that all the parts of the system can be trained jointly.

4

Logo Retrieval System

The difficulties of logo retrieval were highlighted in the introduction, in chapter 1. Nowadays, deep neural networks are utilized to solve such difficult computer vision problems. In this work, end to end neural network solutions will be used, to retrieve logos from videos. The set of logos to be searched is called the query set. Furthermore, the frames of videos, where the logos should be retrieved from, is called the search set.

Section 4.1 details the publicly available logo datasets and introduces a self annotated dataset from sport videos. Section 4.2 shows how Faster R-CNNs can be utilized as logo detectors. Furthermore, section 4.3 presents the different possibilities how logo image correspondences can be found in a dataset. It introduces also a solution which includes both tasks in one network. The chapter is concluded with a description, how a video is processed, and how the results are calculated.

4.1 Logo Datasets

The hunger of deep learning methods for training data is well-known. This section introduces the datasets, used in this work. First, section 4.1.1 details the publicly available logo datasets. Since these datasets are relatively small, a better training result can be achieved, if data from the target domain is involved too, and then the datasets are merged. This additional dataset is presented in section 4.1.2.

4.1.1 Public Datasets

The different logo datasets with the number of brands, images and bounding box RoIs can be seen in table 4.1. The total number of brands means the number of different brands altogether.

	Number of brands	**Number of logo images**	**Number of RoIs**
BelgaLogos [Jol09], [Let12]	37	1,321	2,697
FlickrBelgaLogos [Let12]	37	2,697	2,697
Flickr Logos 27 [Kal11]	27	810	1,261
FlickrLogos-32 [Rom11b]	32	$70 \cdot 32 = 2{,}240$	3,404
Logos-32plus [Bia17], [Bia15]	32	7,830	12,300
TopLogo10 [Su16]	10	$10 \cdot 70 = 700$	863
Total	**80 (union)**	**15,598**	**23,222**

Table 4.1: Publicly available logo datasets with bounding box annotations

There is also a trademark dataset available having a much greater cardinality, called METU Trademark [Tur17]. However, the images of this dataset contain only the logo of a company, without any context. This dataset turned out to have no use for region-based deep learning methods, since this approach needs to learn to distinguish between objects and the background. The network was trained with the fusion of FlickrLogos-32 and METU trademark dataset. The training setup is described in section 4.3.5. Afterwards, it was tested with the evaluation method of py-faster-R-CNN [Gir17] [Ren15], described in chapter 5.

However, this training data is insufficient to train a network from scratch (with randomly initialized weights). Girshick et al. showed [Gir13], that initializing the weights of the network from a CNN, which was trained on an unrelated dataset, and fine-tuning on the target dataset, can boost on performance significantly. It is because of the hierarchical learning of shapes by the layers of the convolutional neural network. As a result, the learning of the first several convolutional layers can even be turned off during fine-tuning. Thus, for all the training in this work, the weights of models are initialized from a network, pretrained on ImageNet classification dataset [Den09].

4.1.2 Self Annotated Dataset - SportLogos

For dataset extension and evaluation purposes, there were four additional datasets created by extracting them from sport videos. These data are needed, to be able to fine-tune the networks for the specific context. All the logos of these images were annotated despite occlusion and bad sight of them, along with company name. The collected datasets are summarized on table 4.2. The utilization of the datasets (training or evaluation) are indicated too. The one, used for testing, is from a different TV broadcasting company than the other three sets, and contains logos which are very challenging to retrieve.

	Phase	**Number of brands**	**Number of logo images**	**Number of RoIs**
Football-1	Train	104	331	3,329
Ski		27	179	701
Ice hockey		19	410	3,920
Football-2	Test	40	298	2,348
Total		**143 (union)**	**1,218**	**10,298**

Table 4.2: Collected logo datasets from sport videos

4.1.3 Dataset Fusion

Figure 4.1 summarizes the distribution of the number of logo brands in the fusion of the public and the SportLogos dataset. The latter introduces a large number of classes with low number of examples. Although, these classes are valuable for the training of a logo detector, their usefulness to train a classifier is doubtful. The SportLogos dataset contains some brands, which are already in the set of public datasets. After fusing the datasets with the logos of 218 brands, it already includes the logos of a lot of common sport brands. Thus, it is hard to find a sport video with logos, which cannot be already found in the training set. The figure 4.2 presents the extent of intersections. This means, that four brands are already comprised by the training data, from which two are negligible, because of having fewer than five training examples. The effect of other two classes will be examined further in chapter 5. The list of all brands and the number of contained ground-truths can be found in the Appendix.

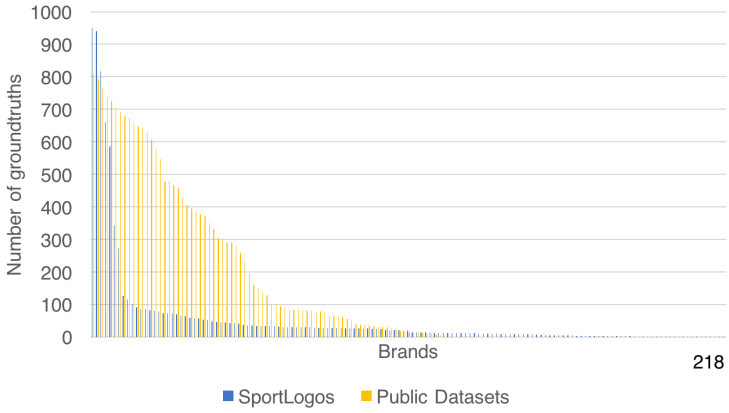

Figure 4.1: Cardinality of different brands in the public and SportLogos datasets

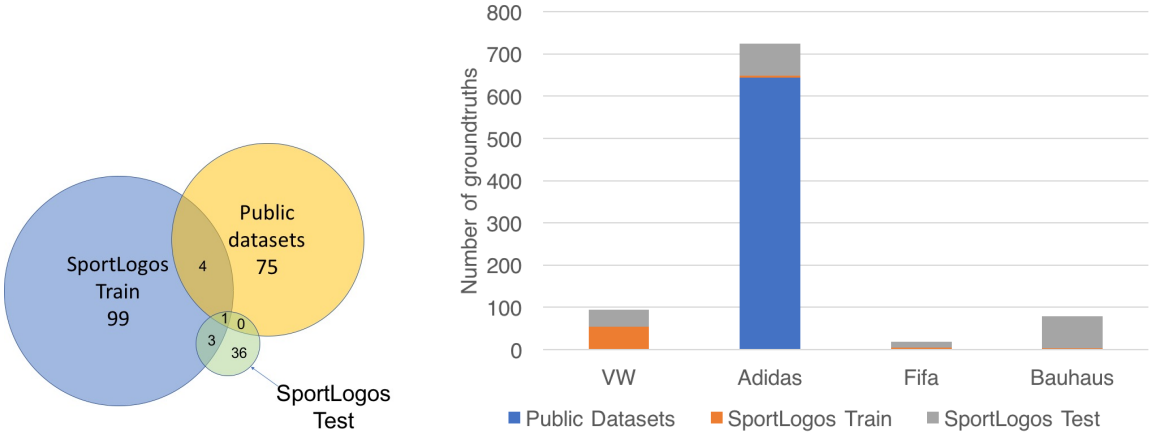

Figure 4.2: Left: Number of classes within the different datasets, and their size of intersections. Right: Brands contained by both the training and the test dataset and their cardinality

4.2 Logo Detection

For scene retrieval it is conventional, that one feature is created for an input image. This is achieved either by inferring from the complete image or by searching for key regions and then extracting features from the found regions, which are finally fused into a global feature. For logo retrieval the goal is not the extraction of a global feature, because it would not be descriptive enough to retrieve small objects. Additionally, a global feature usually does not preserve information of the location and the size of the objects, which are also important factors for logo retrieval.

Therefore, first the objects should be detected in the input image. There are a lot of possibilities to search for objects as explained in chapter 2. Keypoint detectors and external proposal systems are translation and rotation invariant, but usually these systems cannot be trained on a specific dataset. Girshick et al. [Ren15] proposed the Faster R-CNN, detailed in section 3.4.3, for end to end learning to detect and classify objects on an image. This network has a bounding box regressor for each trained class, thus it is capable to produce object type specific region proposals.

In the following sections, different detectors will be introduced.

4.2.1 Region Proposal Network

During the training of a Faster R-CNN, a Region Proposal Network will be trained to detect all kind of objects, which it was trained on. Thus, after training a Faster R-CNN with different logos, the trained RPN can be used alone as a logo detector. It has the advantage, that the detector can be extracted from every already trained Faster R-CNN, unlike those in case of the following solutions.

4.2.2 Class Agnostic Faster R-CNN

Faster R-CNN can be trained for two classes: background and logo. For this purpose the classes of every annotation box will be neglected. This solution is expected to yield better performance than the RPN detector. Firstly, because of the fully connected layers preceding the final classifier and the bounding box regression layers. Secondly, it is a cascade of two detectors (RPN and FC), similarly as in [Vio04], where the RPN is a weaker classifier, which is faster, allowing more false positives. For the combination it is expected to have a lower false positive rate.

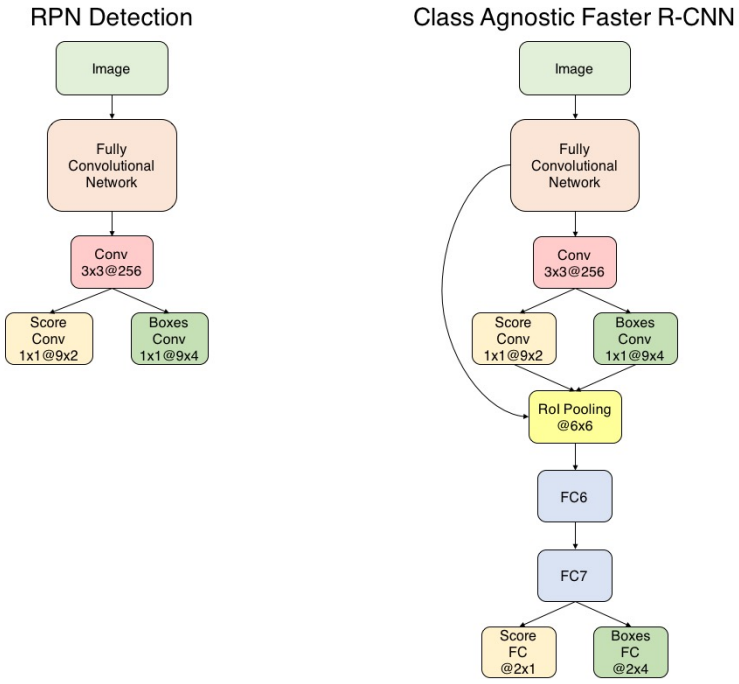

Figure 4.3: Left: Object detection with Region Proposal Network. Right: Object detection with Faster R-CNN. It can be considered as cascade of a weak and a stronger classifier

4.3 Logo Comparison

After a logo is detected, a correspondence from the query set should be found. In order to retrieve as much objects from the images as possible, the detectors should work with a high recall. Although, for difficult tasks like open-set logo detection, high recall values induce low precision, thus a lot of false positive possible object locations are produced. These examples should be eliminated by the classifier.

However, in case of image retrieval the goal is not direct classification, but rather feature extraction of an RoI. Thus, the features of the logos of all the query images and the search set should be collected, which can be solved in several ways, detailed in this sections. The retrieved feature vectors are then

normalized and the similarities of them are calculated either with euclidean distance or cosine similarity. The similarity score of the former is computed by calculating the *L2* (euclidean) norm of difference of the two vectors, then it is placed in the following formula:

$$\frac{1}{1+L2(f_1-f_2)} \tag{4.1}$$

whereas $f1, f2$ are the two features to be compared. The cosine distance is computed by calculating the cosine value of the angle between two vectors. For normalized features the magnitude information of the vector is discarded thus, the two distance metrics yield the same results. The distance is then used as probability score of the correspondence to a query image. The detections on a particular location, not having the largest score, are eliminated by non-maximum suppression, which searches for other bounding boxes with a minimum IoU of 0.3.

4.3.1 Baseline: Faster-Logos

The features of both the query and the search set can be extracted by running a Faster R-CNN on them in a standard way, and utilizing the output vectors of the last or the second last layer as features. The advantage of this solution is its speed. This is the fastest solution among the detailed ones. But there are several drawbacks. It has low performance, because of the unknown classes, especially if the net is trained for a small number of classes. This results in a low dimensional feature vector e.g. 32 after training with FlickrLogos-32 by using the class probabilities as feature which yields often the best descriptor.

The Region Proposal Network outputs several hundred possible object locations (default is 300 in test phase) for every input image. Processing so many query features would immensely increase the computational burden. Thus, it is advantageous to filter the detection list. Although, this cannot be done with the classification scores, because the images contain logos from unknown brands, so the descriptors are rather a collection of different brands. For this purpose, the score output of the RPN can be used, which is an objectness indicator. This score can be adopted for the detections of the search images too. Therefore, one can take the RoI with the greatest score for query images, and set a low threshold for images of the search set. The proposed regions are then adjusted with the bounding box regression of the class with the highest probability score. Figure 4.4 shows the outline of this solution. This solution is chosen for baseline, because it is prevalent in recent research of closed-set logo retrieval in [Bao16], [Oli16], [Qi17]. It is evaluated in section 5.3, and shown that it has almost state-of the-art performance in closed-set logo retrieval. A next issue, which discourages from applying this method for open-set retrieval, occurs during running the network on the query examples. The network often does not estimate the complete logo with the greatest score, but only a part of that (see the Registered Trade Mark symbol in figure 4.5). This mislocalization destroys the retrieval of that entire class.

4.3.2 Fast&Faster-Logos

The drawbacks of the solution in section 4.3.1 imply, that the RPN should be turned off for the examples of the query set. This means, that the network is applied in fast R-CNN mode. For external region proposal, one region is applied, which includes the complete query image. However, it may yield worse descriptors, because of the loosely fitting bounding box. On the other hand, the logo positions and their features of the search set can be inferred with Faster R-CNN, and filtered with a threshold as described in section 4.3.1.

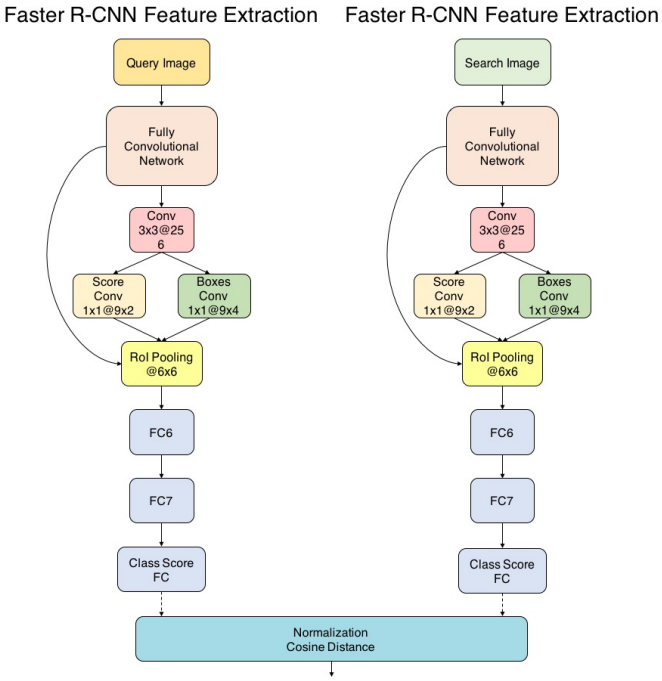

Figure 4.4: Baseline logo retrieval system applies Faster-R-CNN both on the images of the query and the search set, then compares the features

Figure 4.5: Misplaced logo detection, with maximum RPN score

4.3.3 Fast-Logos

The system in section 4.3.1 may suffer from the inability of the RPN to localize unknown logos. The solution can be further improved by using the best logo detector from the section 4.2. The output of the detector is then treated as external proposals, and thus the system runs in fast R-CNN mode for the images of the search set. This setup is very similar to the one of fast R-CNN, detailed in section 3.4.2, but utilizes neural networks instead of selective search as external region proposal system to collect object locations.

4.3.4 R-CNN-Logos

Section 3.1 details the evolution of convolutional networks by going through the most important ones of today. Donahue et al. proposed [Don13], that convolutional networks can produce excellent descriptors of the input image, regardless of the absence of fine-tuning to the specific context of the image. For this purpose, a network is pretrained on very large datasets, after which it can be deployed for a broad set of computer vision problems. But it cannot localize objects on an image. For region proposal purpose, a trained logo detector can be utilized from section 4.2. This setup of networks is very similar to the R-CNN, detailed in section 3.4, but omits the use of selective search as external region proposal system. As such,it has the disadvantage of retrieving features of the regions separately from each other by not sharing the computed feature maps. Thus, it needs much more time, to infer a complete image. On the other hand, it is beneficial for the performance, because the

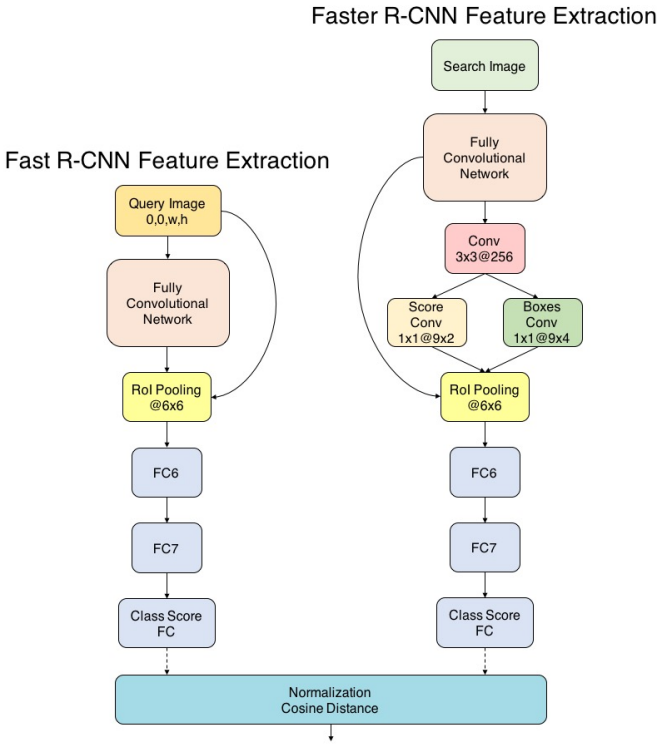

Figure 4.6: Fast&Faster-Logos solution. It computes the query feature from the complete query image (i.e. Fast-R-CNN mode), the features of the search image is collected by Faster-R-CNN

complete network is only focused on a specific region. This part of the system can be easily swapped to reach the desired performance and time constraints, since all the kind of networks, explained in section 3.1 can be utilized for feature extraction. Figure 4.8 outlines this solution.

4.3.5 Siam-Logos

Both a class agnostic detector (section 4.2.2) and a classifier can be built together in a Faster R-CNN and trained jointly. For this purpose, either all fully connected layer or only the two at the end of the network, responsible for classification and bounding box regression should be duplicated. In so doing, one branch will be trained with brand label, the other only with logo object indication. In this network, the weights of the feature extraction layers and the Region Proposal Network can be shared between the two tasks. The easiest way to train such a network is a siamese like Faster R-CNN. A schematic illustration of the training and test setup can be seen in figure 4.9.

Siamese networks [Had06] are basically used for calculating similarity scores between inputs. If it is combined with an appropriate loss function e.g. contrastive loss [Had06] or max margin loss [Sim13][Her16a], the network can be trained with an image pair and a label indicating whether the objects on the images are from the same category or not. The network then learns to project objects from the same category with a low-, otherwise with a large distance to each other, according to a specified metric. This is achieved by sharing the weights of the feature extraction layers.

For Faster R-CNN, the parameters for Region Proposal Network can also be shared. This setup has the advantage, that training of one task can also improve the performance of the other task, which is not currently trained. In particular for logo retrieval, the network can benefit from a bounding box annotated logo dataset, without specific brand indication. To annotate such a dataset much less human resource is needed. A quantitative evaluation can be found in section 6.1.

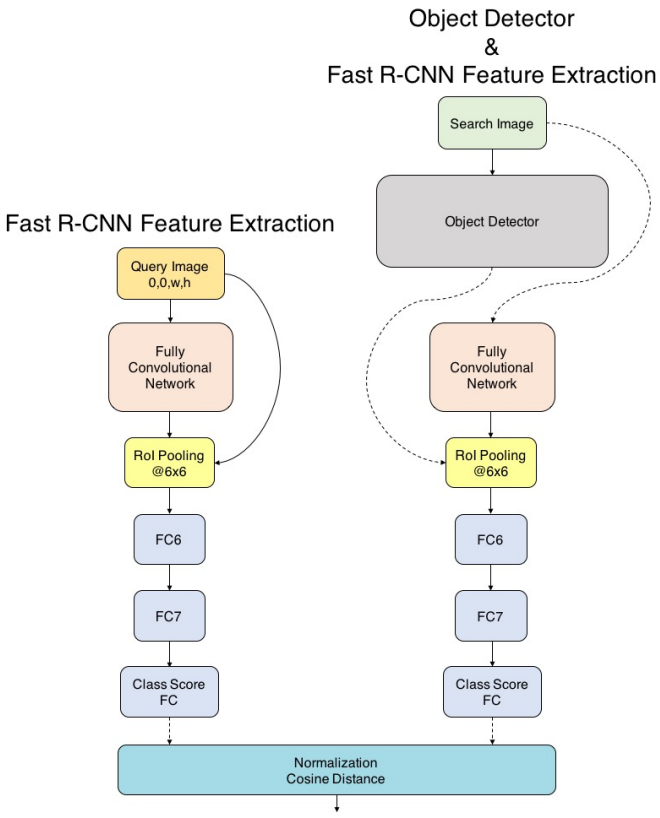

Figure 4.7: Fast-Logos solution. It computes the features both from the query and the search set in Fast-R-CNN mode, whereas the query region is the complete query image and the locations for the search set come from an external Faster-R-CNN logo detector. The dashed lines indicate an indirect connection between the networks.

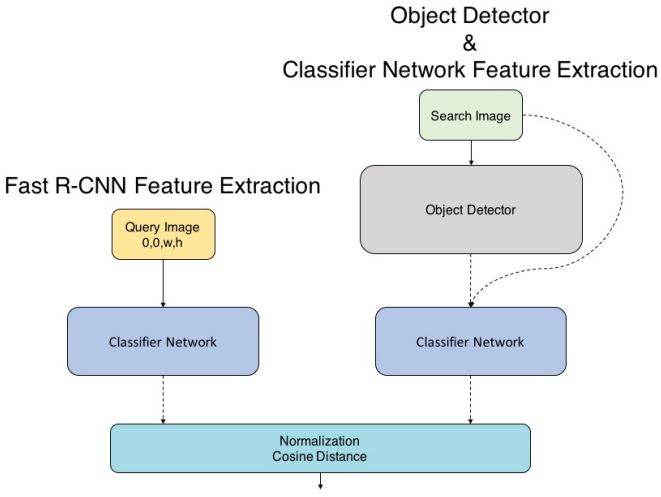

Figure 4.8: R-CNN-Logos solution. Query region is the complete query image, the search region proposals are generated from and external Faster-R-CNN logo detector. The features are extracted with a general purpose convolutional network. The dashed lines indicate an indirect connection between the networks.

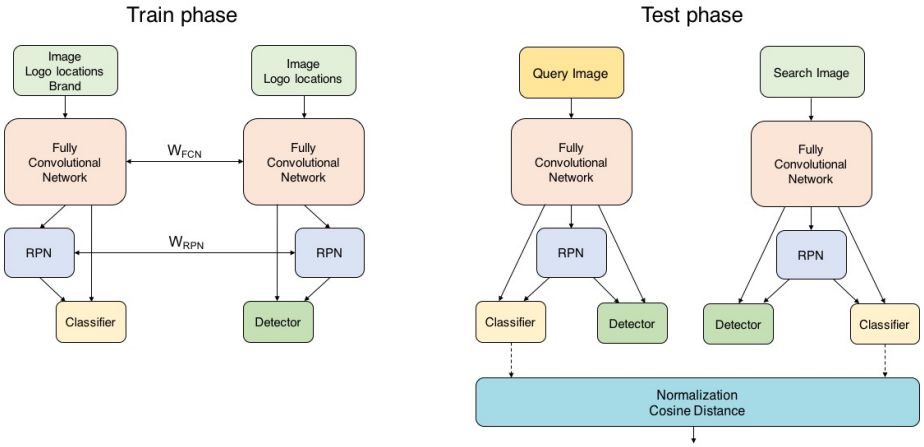

Figure 4.9: Siam-Logos. Network setups in train and test phases for learning detection and classification jointly.

4.4 Video Processing

The purpose of the system is to calculate and summarize for how long the logos of a specific brand has been visible and how large is their cumulative area. For this purpose, the video is cut into frames, and the logos are detected on each frame separately. The visible time is then calculated with the reciprocal of the Frame Per Seconds (FPS) value of the video and summated through every frame. The area of every detection is also calculated and cumulated for all frames. Finally, the pixel resolution of the video is taken, and the cumulated areas are converted into full frame units.

5

Experiments

This chapter introduces some of the experiments, made to prepare for the proposed solutions, detailed in 4. Section 5.1 details the chosen metrics, and common training setups. Synthetic data generation and evaluation will be detailed in section 5.2. Finally, the utilized basic networks, trained on different set of the datasets, will be evaluated on the test set of FlickrLogos-32, which counts as the standard evaluation set in logo retrieval.

5.1 Evaluation Methods

The models trained for logo detection are evaluated, with Free-Response Receiver Operating Characteristic (FROC) curve [Mil69]. This metric was first used for cancer localization in medical images. On this curve the Detection Rate (DR), i.e. the fraction of the number of the true positive detections and the number of all the positive locations in the dataset, is plotted over the average number of false detections per image. Since the detectors should be optimized to have a recall, as high as possible, this curve gives an intuitive interpretation about the performance of a detector.

Precision and recall are favoured values in image retrieval. Precision is the fraction of the number of relevant retrieved objects and the number of all the retrieved objects. Recall is the ratio of the number of relevant retrieved objects to the number of all the relevant objects. Although, usually a retrieval system is capable to return a ranked list of the retrieved objects, precision and recall ignore this information. In order to take these lists into account, the very popular mean Average Precision (mAP) is calculated. Firstly, a ranked list will be constructed, then the precision curve will be gathered as a function of recall. The average precision is computed as the area under this curve. The average of these values gives the mean average precision.

The published results are calculated with the evaluation implementation of py-faster-R-CNN [Gir17] [Ren15]. Firstly, it creates a descending sorted list for every company (class), based on the probabilities of being logos from the specific brand on given positions of the images. The lists are then used either to count the true and false positive (TP, FP) detections, which is the base of the FROC curve, or to acquire the precision curve to calculate the mAP.

The TP and FP values are collected by moving the decision boundary threshold on the scores of a region being object or not in the interval [0.01; 1), with 0.01. The gathered values, TP and FP for every threshold, are normalized with the number of ground truth locations and the number of images respectively, which gives the points of the FROC curve.

If not stated otherwise, all the models are trained for 80k iterations with a base learning rate of 0.001, which is reduced by ten after every 50k iteration. All the training and testing are performed in Caffe deep learning framework [Jia14].

5.2 Training with Synthetic Data

In this section, the effect of synthetic data to the logo detection performance will be examined. For this purpose, the detector was trained on existing and generated datasets.

Synthetic data was already used to test its impact to logo retrieval by [Su16]. However, they reported the performance improvements by extending a very scarce real training data with synthetic data (10 training images of FL-32 pro class). It is questionable, whether the synthetic images would have helped so much, if more real training data had been used (e.g. 40 images pro class by using the validation set of FL-32 too).

5.2.1 FlickrBelgaLogos Dataset

A dataset, which is annotated manually, may contain logos, which stay unannotated. If a system is evaluated on such a dataset, and detects the unannotated logo, it counts as false positive. Thus, a synthetic dataset, called FlickrBelgalogos [Let12] is created for evaluation purpose by pasting the logo annotations of the dataset BelgaLogos [Jol09] on images from Flickr to random positions. One could argue with the correctness of evaluating a detector with this dataset, because alone the contrast difference may make the logos easier to detect on these images.

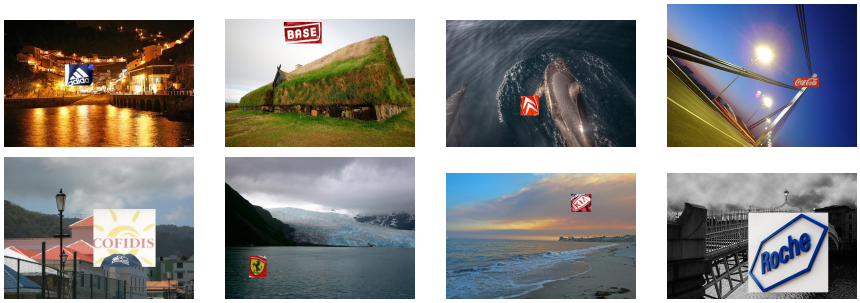

Figure 5.1: FlickrBelgaLogos examples

This dataset was evaluated for training purposes. Therefore a small subset of BelgaLogos was chosen as test set. The logos on these images were left out from the FlickrBelgaLogos dataset. The rest of the images is used as the train set to train a Faster R-CNN model with the 37 classes of BelgaLogos. The trained models are tested on the chosen test set of BelgaLogos. Then the train set of BelgaLogos was also trained with the same network, to compare the results. After that the datasets were fused to examine the possibility of achieving better performance. Lastly, the model was trained with Curriculum Learning [Ben09] (CL) as it was done with logos in [Su16]. CL is a learning process, whereas the examples became gradually more difficult during training. In this context, it is realized by training the network first merely with synthetic logos and then with real images.

The synthetic dataset alone could achieve moderate results, the obvious advantage of a real dataset can be seen in figure 5.2. Unfortunately, the fusion of datasets does not yield extra performance, neither with a simple fusion, nor with Curriculum Learning. The latter has the advantage of achieving convergence much earlier, compared to other training scenarios. In [Su16], 10% relative extra performance was achieved by training with CL, to recognize a restricted number of classes. In case of

FlickrBelgaLogos, this expected to be unsuccessful, because the same logos of BelgaLogos are reused in FlickrBelgaLogos. This means, while trying to achieve better performance, the transfer of logos to another context does not give additional information. As base network the `VGG_CNN_M` [Cha14] was chosen, pretrained on ImageNet [Den09], because of its much shorter training times, compared to VGG-16 (about a quarter as much time), still having a performance good enough for the experiments.

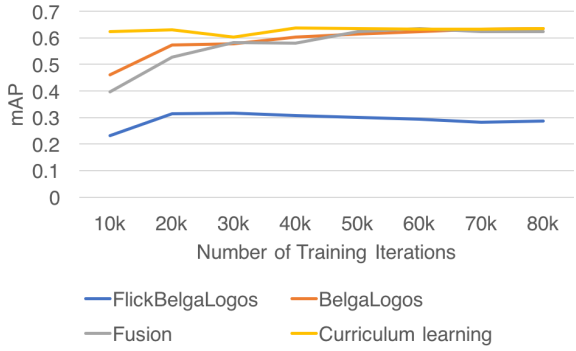

Figure 5.2: Logo recognition performance after training with real (BelgaLogos) and synthetic data (FlickrBelgaLogos)

After that, the open-set logo detection capability of the model, trained only on FlickrBelgaLogos, was tested. A Faster R-CNN was trained purely for logo detection, without logo classes. For evaluation, a self annotated dataset of a sport video was used, that has logos merely from such companies, with which the net has not been trained before. Despite the small size and the unreality of this dataset, it is able to generalize from the learned logos and detect some of the logos unknown for the network. The detection performance evaluation can be seen in figure 5.3 as well as an example detection on figure 5.4.

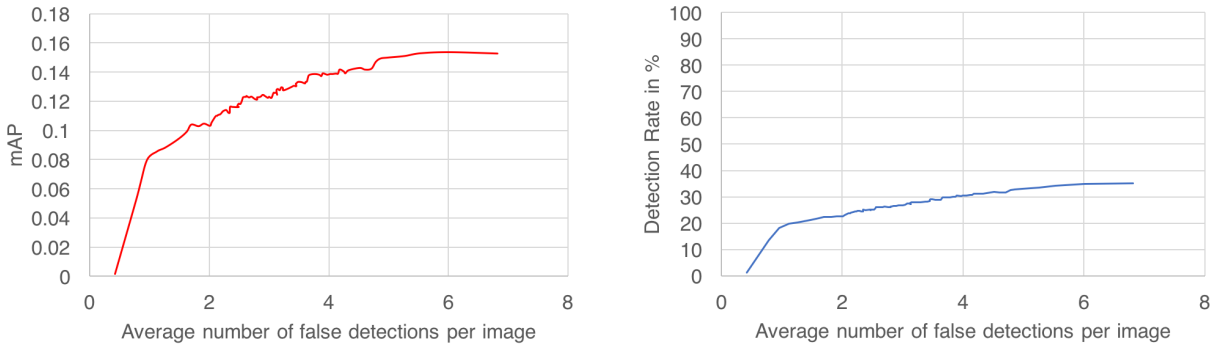

Figure 5.3: Logo detection performance while training purely with synthetic data

5.2.2 METU Trademark Dataset

In order to extend the training dataset, a synthetic dataset was generated, where one logo from the METU Trademark dataset [Tur17] was placed on an image. As basis, photos from Tripadvisor were used to fulfill the requirement of being practically logo-free. The majority of the logo's background has a white color. But logos usually have some background other than white. Therefore some transformations were applied on the logo images before training as follows. One third of the dataset was left unchanged. The brightness of the rest of them was adjusted to the brightness of the image, on which the logo is placed. Furthermore for one third of the logos, the mean HSV value of each logo

Figure 5.4: Logo detection example by a detector, trained only with synthetic data

was calculated and the hue value is rotated with 90 degree. In addition, Gaussian blur was applied on the edges of the logos, in order to suppress the sharp contrast changes. The table 5.1 summarizes the applied transformations.

Logos	Hue rotation	Brightness adjustment
33%	no	no
33%	no	yes
33%	yes	yes

Table 5.1: Applied transformations

The created dataset is then used to train a two class Faster R-CNN for logo detection. The base network is again `VGG_CNN_M`, as in section 5.2.1. For evaluation, the same self annotated sport video dataset was used, as in section 5.2.1. Unfortunately, the trained network was incapable of detecting logos. The problem is most likely because of the dataset, which consists mainly of logos with a white background and a black text on them. The other source of problem could be the unreal appearance of the logos or the transformations applied to them. Figure 5.5 shows some generated examples.

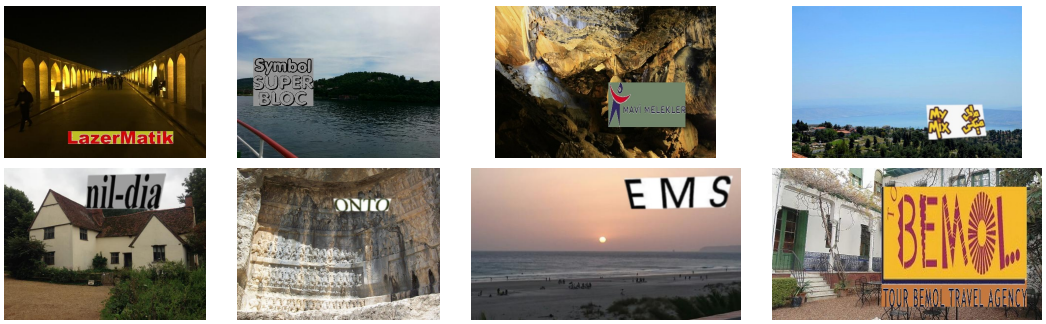

Figure 5.5: Generated synthetic logo images based on the METU Trademark Dataset [Tur17]

5.2.3 Synthetic Data with Shape Based Logos

As section 5.2.2 shows, the most of the text based logos are incapable to train a model to detect real logos. In order to successfully extend the training dataset, logo images with more shape and color were collected from the Logo API of Clearbit [Log17]. The logos were simply copied and pasted to the same Tripadvisor dataset, as in section 5.2.2. Unfortunately, also this dataset was not able to recognize any real logos. As a next trial, the white background of the logos was set to transparent, as suggested in [Su16], but in the context of open-set logo detection, it does not appear to be useful

either. The reason for the success of FlickrBelgaLogos, seen in section 5.2.1, was most likely because the logos and their direct surroundings come from a real scene, and the logos were from the same closed-set of brands.

Figure 5.6: Generated synthetic logo images based on logo images collected from the Logo API of Clearbit [Log17]

5.3 Evaluation on FlickrLogos-32

The majority of logo retrieval research uses the test set of FlickrLogos-32 (FL-32) for evaluation. Therefore, the effect of additional data and different architectures to the performance was examined on this dataset. The trained networks are the following ones:

1. A Faster R-CNN, with `VGG_CNN_M_1024` as base network, trained on the training and validation set of FL-32.
2. A jointly trained detector and classifier, as introduced in section 4.3.4, whereas the impact of further semi-labeled data is examined by training the classifier the same way, as in point 1, and the detector on every other public dataset without specific brand label.
3. A faster R-CNN is trained, based on `VGG_CNN_M_1024`, on the same dataset as the detector and classifier together in point 2, but this time with complete brand specification.
4. Point 1 is repeated with VGG-16.
5. Point 2 is repeated with VGG-16.
6. A Faster R-CNN, trained on all public datasets, and annotated sport videos.

The test set of FL-32 is evaluated with the configurations from points 1, 2 and 3, and compared in figure 5.7. The networks are evaluated based on the mAP. It can be seen, that additional logo dataset, even without brand indication, can be utilized, to detect and recognize known logos with better performance by using the Siamese-like architecture, proposed in 4.3.4. The network, trained on complete class information, begins with a worse performance due to the higher number of iterations required to distinguish in the set of learned classes with a significantly higher cardinality. Later on, it has obviously superior performance.

The improvement of additional data on the performance is then tested with VGG16 base network, having more capacity. Due to more data, it achieves naturally state-of the-art performance on FL-32 test set with 90.2 mAP, whereas the best published result is 84.2 mAP, proposed by [Bao16]. Figure 5.8 shows the performance with VGG-16 base network, compared with the best system using `VGG_CNN_M_1024`, as base network.

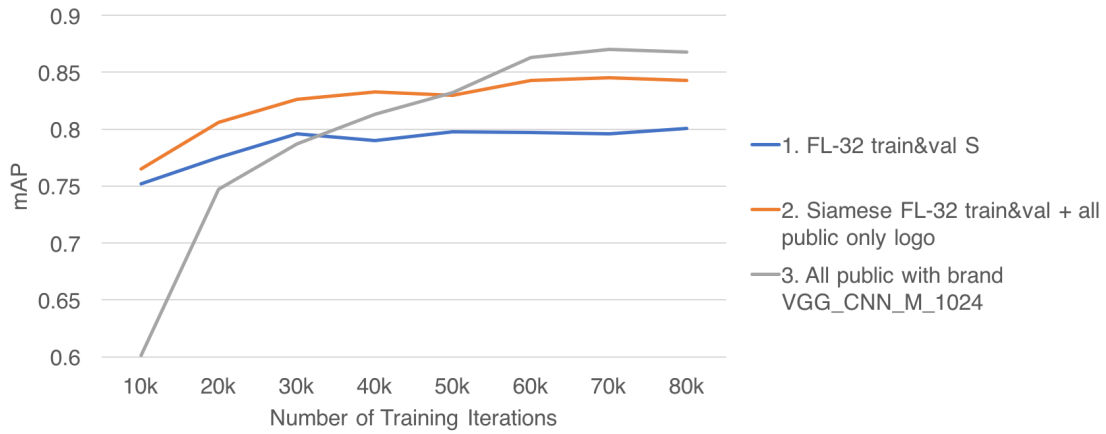

Figure 5.7: FL-32 test evaluation with `VGG_CNN_M_1024` base network

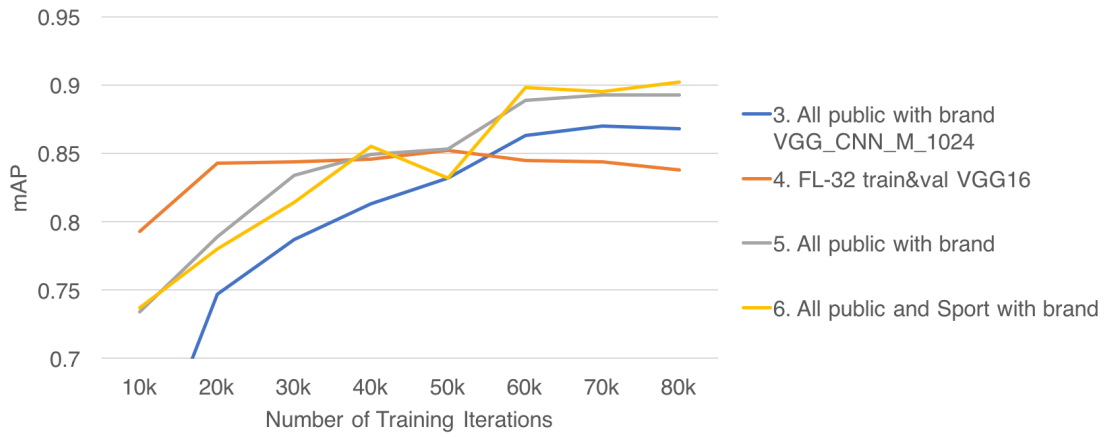

Figure 5.8: FL-32 test evaluation with VGG-16 compared with the best `VGG_CNN_M_1024`, as base network

6

Logo Retrieval System Evaluation

This chapter evaluates the different solutions to detect and retrieve logos from images, detailed in chapter 4. Section 6.1 compares the performance of different detectors and chooses the best performing for further usage. After that, the complete systems are evaluated in section 6.2. Finally, the effect of training of the networks with classes, which are contained by the test set, will be examined.

In order to get a holistic view on the performance of the complete retrieval system, the different solutions are evaluated by plotting the Detection and Identification Rate (DIR) as a function of average number of false positive classifications. DIR can also be considered as a multi-class recall, it involves not only the quality of the object detector, but also the classifier. In order to incorporate the probability scores of the detected logos, with the aim of reducing false positive recognitions, the final similarity scores between query and proposed regions are multiplied by that score.

6.1 Logo Detection

Although, there is not any known earlier work in the field of open-set logo detection and retrieval, it was still attempted to take a solution as baseline, which is already published. As chapter 2 details, there exists a lot of research in improving the retrieval performance on the FlickrLogos-32 dataset using Faster R-CNN. Thereby, the region proposal network of a Faster R-CNN is chosen for logo detector baseline, which is trained only on the train and validation set of FlickrLogos-32. The network has almost state-of-the-art performance in closed-set logo retrieval, on the test set of this dataset. In particular with VGG-16 as base network, it yields 83.7 mAP, whereas the best already published result is 84.2 mAP proposed by [Bao16], using the multiscale Fast R-CNN approach and AlexNet, as base network.

All the results of the following experiments are compared in figure 6.1. The data points are gathered by moving a threshold value on the region objectness probability emitted by the network. Firstly, the RPN of the baseline network is evaluated on the Football-2 dataset, introduced in section 4.1.2. Afterwards, the network is trained on all the publicly available logo datasets with bounding box annotations, introduced in section 4.1. Finally the improvement of the RPN network is tested. This network has already a better recall performance, but it retrieves much more false locations for lower threshold values. Next, a class agnostic Faster R-CNN is trained again on all public datasets, but now only with "logo" class. Both the RPN and the classifier with regression layer at the end of the network are tested. The latter option is referred as "FC" on the figure 6.1, because of the fully connected layers accomplishing the classification and bounding box regression. A quite interesting result is here, that by training the same model on the same dataset but without specific brand labels, the detection

performance of the region proposal network improves (see "Public Datasets RPN" versus "Public Datasets Class Agnostic RPN").

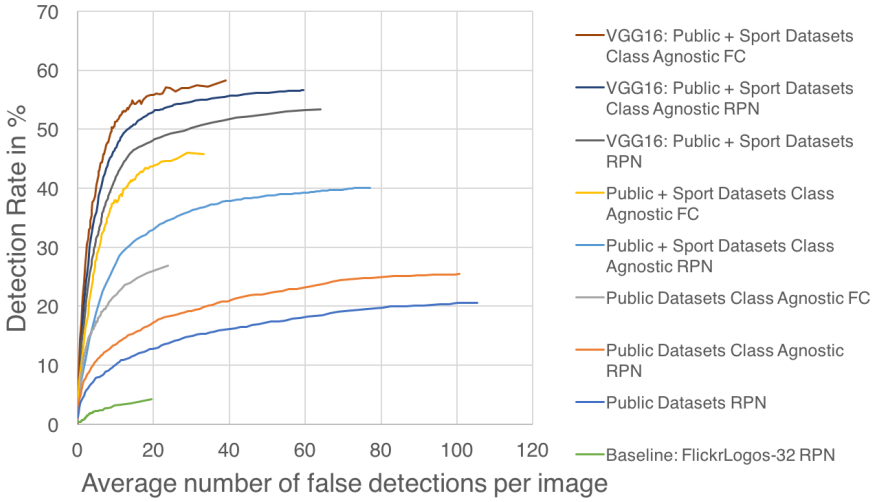

Figure 6.1: Logo detection evaluation

In order to fine-tune the networks for the specific task, the datasets from similar domains were used, introduced in section 4.1.2. The training is saturated after about one epoch of learning. This means, that the initial network is good pretrained for the task. This data has a large effect on the performance.

Unfortunately, the test dataset contains a relatively small number of brands, which are already contained in the fused training dataset. Nevertheless, the effect of that is questionable. E.g. Adidas is one of the brands, which occurs both in the training set and with a large number on the video. On the other hand, Adidas logos are also included in the datasets FlickrLogos-32 and Logos32Plus and the networks, trained already on these datasets achieve much weaker results.

Finally, a VGG-16 based Faster R-CNN was trained on all the training data, used earlier. Although, the gap between the RPN and the FC logo detector of this network became smaller, due to the much deeper base network before the RPN, the FC solution still yields superior performance both in recall and precision. One could argue, that the marginal improvement of FC to RPN does not worth the increased computational complexity. However, the RPN network has about double as much false positive on the same level of recall. This can cause much more additional computational burden for the classifier than what the FC layers do, especially if it needs large computational load.

The VGG-16 detector can generalize quite well, as figure 6.2 shows, it is able to operate under extreme light conditions too.

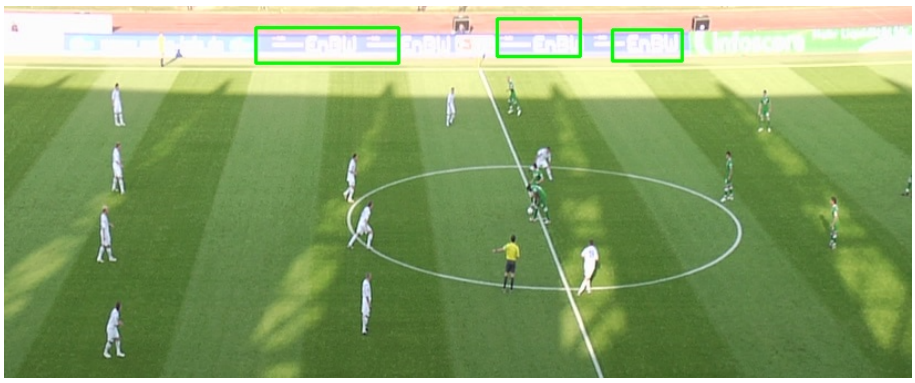

Figure 6.2: Logo detection under extreme light conditions

6.2 Logo Retrieval

After the best logo detector is found, the system should decide, which detections are relevant, and classify them, to the instances of the query set.

As a baseline system for logo retrieval, the same solution was chosen as for logo detection baseline - section 6.1, which is the Faster-Logos solution, detailed in section 4.3.1. It consists of a Faster R-CNN, trained for logo detection on the train and validation set of FL-32.

Nevertheless, this solution may be incapable of detecting the complete logo, especially if the logo has more distinct parts, as detailed in section 4.3.1. The effect of mislocalization and the low size of output feature vector of the last layer is further investigated. To this end, the performance of the class probabilities was qualitatively tested. In particular, three logo images were tested from the same, unknown brand, having lesser or greater appearance variation:

- a logo, cropped from a video,
- a logo with very similar background color, but in high resolution,
- a logo with a completely different color scheme also in high resolution.

The logos fill out the majority of the images, as figure 6.3 shows. The assumption was, that despite of the unknownness of the brand, the descriptor features of the three logos will contain similar portion of probabilities of known brands, thus having low distance to each other. Unfortunately, the network was incapable to yield such feature vectors. The original logos and the most likely predicted logos from the known set can be seen in figure 6.3. The composition of the normalized features for the three query images is shown by figure 6.4.

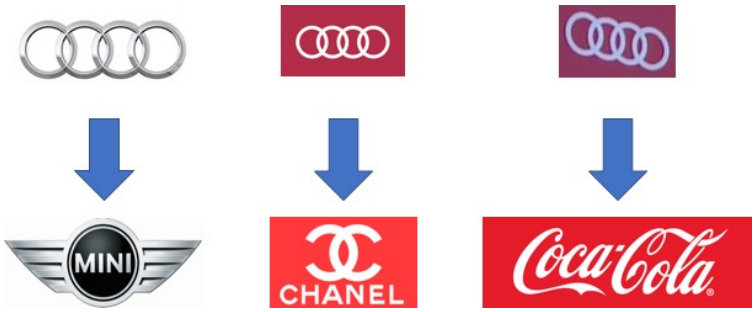

Figure 6.3: Misclassification due to the small number of trained classes and data

As section 4.3.2 details, the performance of this system may be improved by inferring the complete query image, instead of its most probable logo location. The system is evaluated on the Football-2

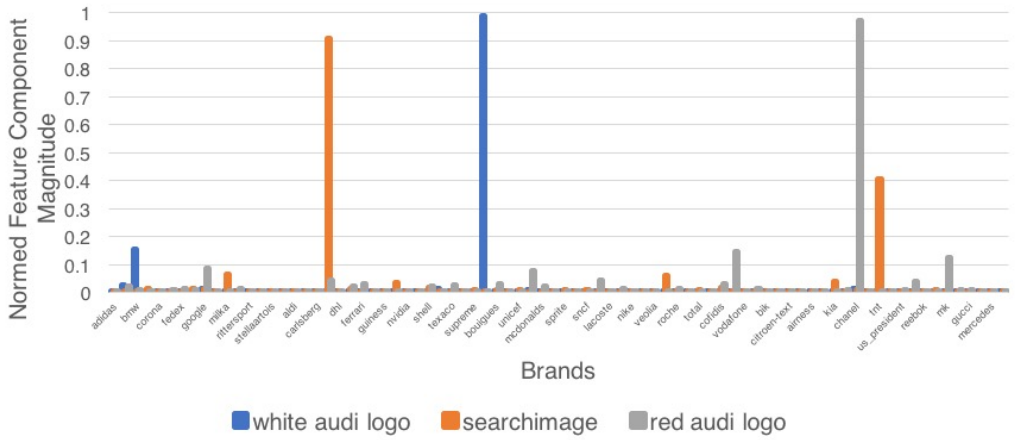

Figure 6.4: Feature vectors of the two query images and one cropped image from a video

dataset, introduced in section 4.1.2. For query set, logos from the videos were cropped. Because of the large intraclass variance shown in figure 6.5, multiple instances from some classes were chosen. But it is still a challenging task for the classifier, because of blur and different viewing angles of the logos in the search set.

Figure 6.5: Samples for intraclass variation in sport videos. Images in the same columns have the same class.

Occluded images are not included in the test set, because perhaps none of the TV-viewers would find the correspondence for unknown logos. Images, which have an edge size both in width and height smaller or equal than 25 pixel, are also omitted, because those very likely escape the viewer's attention. In addition, classes, having only one example, were removed, because in this case the query set would completely cover the ground truth set.

The similarities between the extracted feature vectors were calculated by cosine distance, and the output of the second last layer appeared to perform best as feature. The system is trained on all publicly available datasets. The low performance of the system is primarily induced by the classifier, which is weak in feature extraction from images with unknown classes. Thus, the network was additionally trained on the SportLogos dataset. This self annotated dataset induces a lot of new classes with very few ground truths, as shown in section 4.1.2. Their instances may not be enough to completely train the classifier for those classes. Thus, it was questionable, whether the performance of the network enhances, if it is trained additionally also on the sport dataset. As figure 6.6 shows, it yields better quality. It will be examined in the next paragraph, whether it is only due to the improvement of the RPN, or the classifier extracting better features. The performance of a random classifier is also indicated. This classifier assumed to work similarly as a real one, by outputting a class and a probability score for every region. Since it gets the ground truth regions as detections, it has a lower number of false positive classifications at the maximum of its recall, which is 2.5% (there are 40 classes in the test set).

The jointly trained classifier and detector, introduced in section 4.3.5, did not seem to be useful

Figure 6.6: Performance of Fast&Faster-Logos compared with the Baseline and a Random Classifier

for the case, if the ground truths are completely labeled with brand indication. Both branches were trained with the entire logo datasets, with and without brand label respectively. It achieved lower performance than the Baseline solution, which is trained on the same dataset. Figure 6.7 shows the performance of this solution. As shown in section 6.1, a region proposal network may have inferior

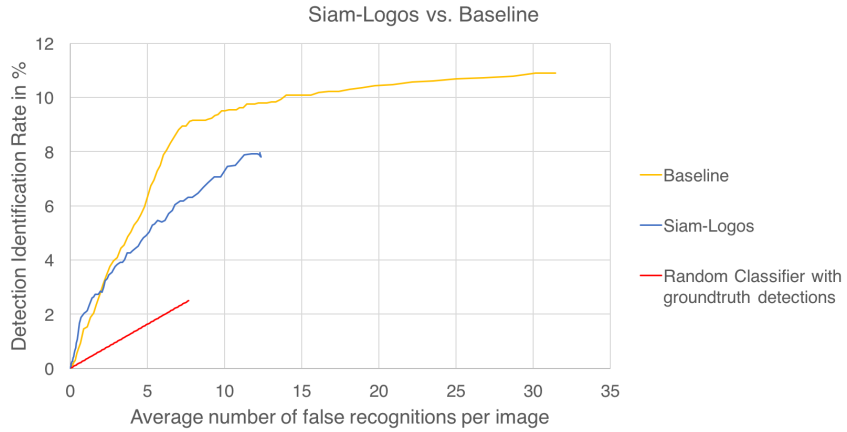

Figure 6.7: Performance of Siam-Logos solution compared with the Baseline and a Random Classifier

performance in logo detection compared to a class agnostic solution. Thus, the best performing FC logo detector can be applied, to ease the task of the classifier by yielding much fewer false positive detections. The features from the detections were then extracted with the same two networks as in the last paragraph, but now applied in fast R-CNN mode. Figure 6.8 shows, that the system has a slightly better performance, but the classification power of the two networks is the same. This means, that the SportLogos dataset helped only in the detection task.

In order to achieve better classification performance, a feature extractor neural network can be placed after the logo detector, which is the idea of R-CNN, as shown in section 4.3.4. The proposed bounding box regions are warped to 224x224 pixels, which is the conventional input size of these networks. No padding was used for the crops, so the aspect ratio was not preserved. However, this should not induce errors, because the query images undergo this aspect ratio change too. There were several feature extractor networks tested, and pretrained on ImageNet dataset by utilizing either the classification probabilities or the output of the second last layer. Figure 6.9 summarizes their performance. Interestingly, the performance of the pretrained `VGG_CNN_M` network surpasses the pretrained ResNet. Additionally, there were two networks trained for classification on the data,

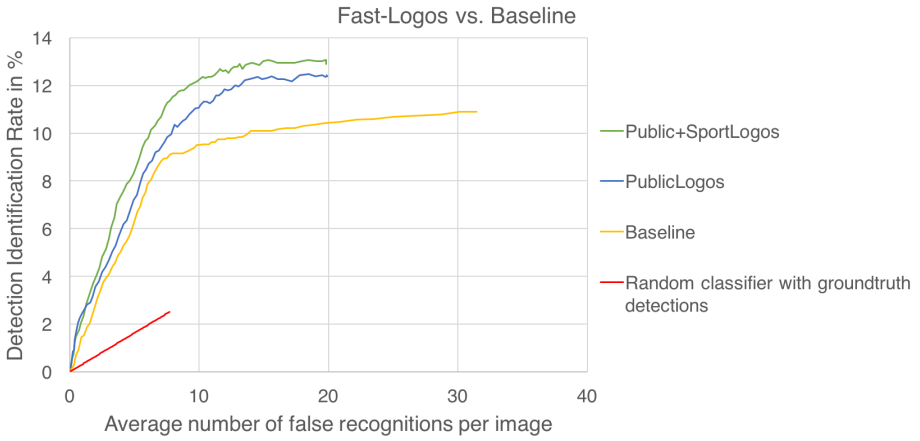

Figure 6.8: Performance of Fast-Logos, i.e. replacing the RPN of a Faster R-CNN with a logo detector Faster R-CNN

cropped from all the logo datasets, preserving 10% of it for validation:

- A network [Her16b] which is optimized for low resolution face recognition (32x32) thus, having the lowest inference time, among the evaluated feature extractor networks. This network could be utilized for logo recognition too, because of the logos, having low resolution on the videos. This network reached a classification accuracy of about 60% on the validation data during training. The network was trained from scratch.
- A ResNet-50 network with reduced input size of 112x112. For this purpose, the last (5.) group of ResNet blocks was removed. This resolution is chosen because the network with input size of 224x224 was not able to converge. The reason for that could be the training data, having a much lower original average resolution of 137x102 (std.dev. 131x102). A network with this size is advantageous for the test dataset too, which has an average ground truth size of about 80x36 pixels (with std.dev of 47x20). Moreover, it is favourable also for the computation time. The weights of the network were initialized from the ImageNet pretrained ResNet-50 network. After training, it achieved 92% accuracy on the validation set.

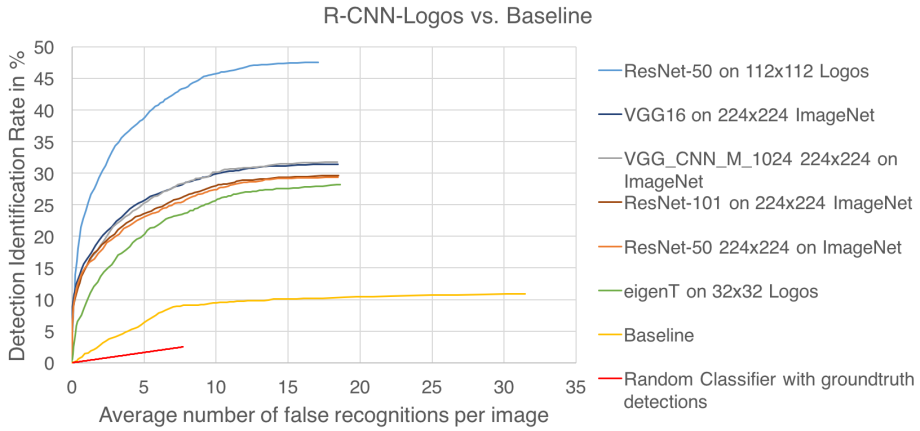

Figure 6.9: Performance of the R-CNN-Logos solution by applying Faster R-CNN as region proposal system

6.3 Effect of Pretrained Classes

Since the training dataset already contains some classes, which are part of the test dataset as shown in section 4.1.3, the effect of this data had to be examined. For that purpose, the best performing networks were taken into account, one pretrained on ImageNet, the other on all the logo datasets. Then, the DIR curves are plotted for the two intersecting classes, with significant examples in the training set. The curves are plotted this time with the number of false positive recognitions. The figure 6.10 shows the result of the experiment, whereas the performance on all the classes is plotted too, over the number of false positive recognition per brands for comparison reasons. The following conclusions can be drawn:

- Pretraining with classes from the test set is less influential, as expected:
  - Converging point of Adidas without pretraining (FP: 4) DIR = 36. With pretraining DIR = 49.
  - Converging point of VW without pretraining (FP: 27) DIR = 55. With pretraining DIR = 60.
- These two classes were much easier for the classifiers to retrieve, regardless the pretraining on the same classes.
- Although the false positive recognition with class pretraining is lower than the average, it is not significant.
- The number of false positive recognitions of the ImageNet network are much lower for low similarity thresholds. This could mean, that the network tends to choose pretrained classes for the detections, which could explain the additional misclassifications.

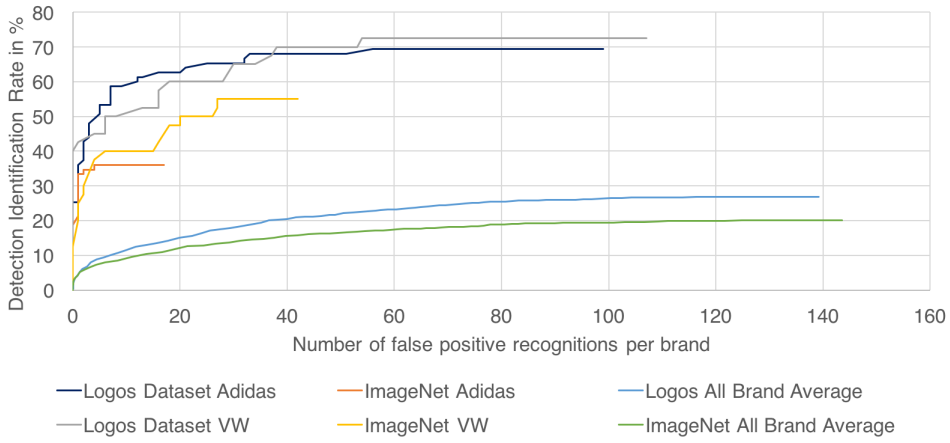

Figure 6.10: Effect of pretrained classes. The performance of a network, trained with and without the two classes, occurring in the test set, and the average performance of the networks.

6.4 Perfomance evaluation

To evaluate the feature extraction performance of the best system the output of the system is examined in detail. For this purpose, the output of the feature extractor network (the output of the last layer before softmax layer) is calculated for a crop image from the search set, which was misclassified, the query images of the real class, and the query image of the class, which was inferred incorrectly. The crop region was emitted by the detector network. The figure 6.11 shows the crop from the query image in the middle, and the other query images around them. The similarity scores are indicated

on the lines between the images, which was calculated with cosine distance. It can be seen, that the wrong query image has the greatest similarity score.

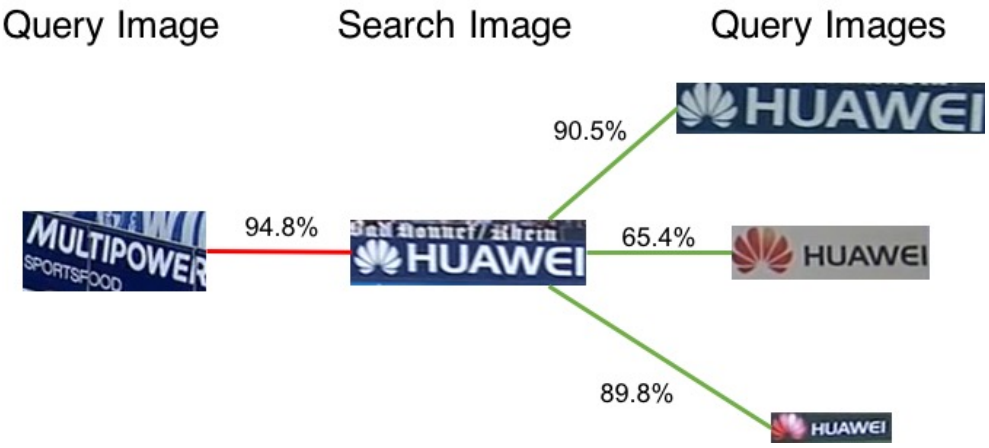

Figure 6.11: Similarity scores between query images of two classes and a crop image from the search set (middle).

6.5 Retrieval times

For the retrieval system not only the performance should be taken into account, but also the time needed to retrieve images. The average retrieval time per search image of every solution is measured and outlined in figure 6.12 together with the achieved highest DIR value. However, these data may

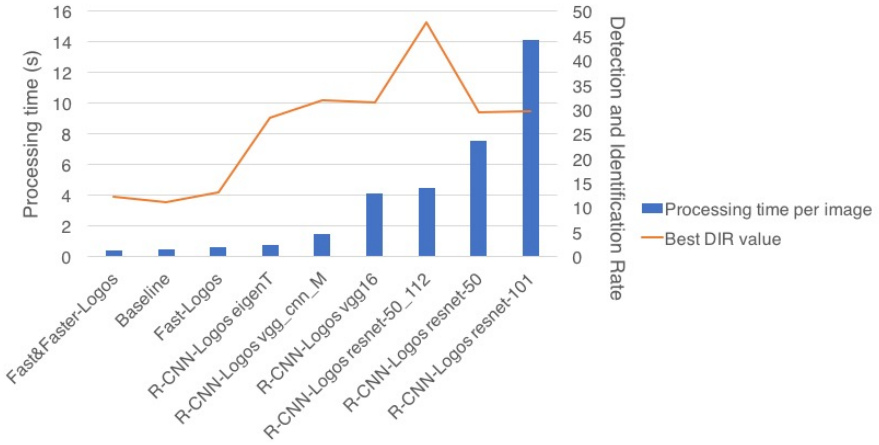

Figure 6.12: Average processing time per image and the best achieved DIR value

not necessarily help to select the right system, according to the performance and time requirements. In order to combine the performance of the network and the average retrieval time, it was calculated, how much DIR value can be retrieved in one second with the different solutions. This evaluation was only accomplished for the R-CNN-Logos solutions, since these outperformed the other proposed systems with a large margin. The figure 6.13 shows the result of this evaluation.

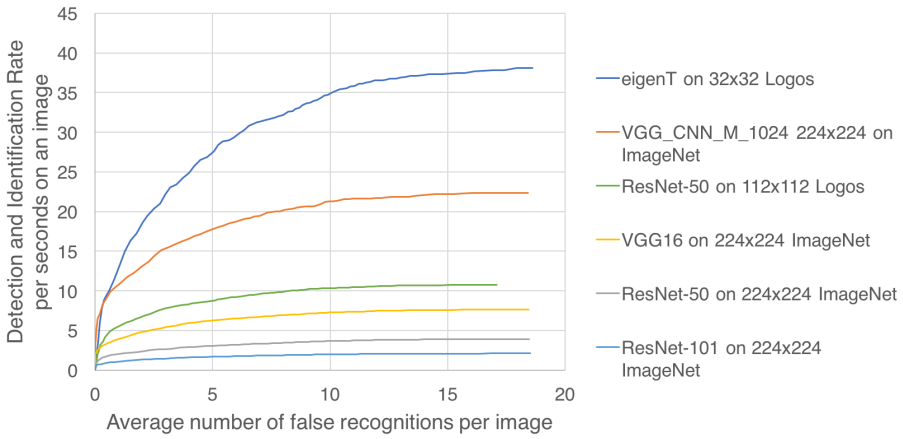

Figure 6.13: Incorporating retrieval time and performance of the different networks of the solution R-CNN-Logos

7

Conclusion

7.1 Summary

In this work, a deep learning based open-set logo retrieval framework was introduced, in the scope of sport videos. The task has a lot of challenges, which is due to the often poor visibility of the logos in these videos. The proposed system rests upon proposal based Convolutional Neural Networks. The region proposals are generated by a Faster R-CNN, trained for detecting logos generally. To compare the quality of the proposed system, a Faster R-CNN is taken as baseline, which generally holds almost state-of-the-art performance on closed-set logo retrieval, evaluated on the test dataset of FlickrLogos-32. The achieved performance of the introduced solution surpasses the baseline method by a large margin.

7.2 Future Work

During this work, it was realized, that the majority of logos in sport videos are text based. This is especially prevalent for videos from small sport events, wherein the advertisements are from local, not so well known companies. Additionally, one should consider the figure 7.1, which would very likely fool the proposed logo detector with the sign "TRIBUNA", resulting in a false positive detection. For this reason, a logo retrieval system should be extended by a text recognition subsystem e.g.

Figure 7.1: A tough example for the logo detector, because of the sign on the top left part of the image

with the work of [Jad14]. It utilizes synthetic data to train a neural network to recognize words. For unknown words, the system could generate training data, and fine-tune the pre-trained network on demand. For this purpose, the needed computational time should be examined. However, there are a large number of logos, consisting only of a shape (e.g. Nike, Chanel, Audi, Apple, etc), and there are

font types, differing significantly from the common fonts. These circumstances would make the task of the text recognizer difficult. Therefore, the system should be able to detect and recognize text on the query image, then search for correspondences in the search set only by text recognition or with a combination of convolutional feature similarities as done in this work. The performance of the feature extractor part of the system can be further increased by building an ensemble of them with an appropriate combination.

Maximum Activation of Convolutions (MAC) was successfully used by [Tol15], [Azi14], [Rad16] for object and scene retrieval. It could be evaluated for logo retrieval too as another feature extraction method.

In order to increase the robustness of the system and reduce the false positive recognitions, the logos, which are detected only for a few frames can be discarded. Additionally, if a logo is recognized many frames in a row, the integration of an object tracking solution could help to identify and fill gap frames, wherein the particular logo is not detected. Such a system could reduce the number of misclassifications further.

Bibliography

[Azi14] AZIZPOUR, Hossein; RAZAVIAN, Ali Sharif; SULLIVAN, Josephine; MAKI, Atsuto und CARLSSON, Stefan: From Generic to Specific Deep Representations for Visual Recognition. *CoRR* (2014), Bd. abs/1406.5774, URL http://arxiv.org/abs/1406.5774

[Bab14] BABENKO, Artem; SLESAREV, Anton; CHIGORIN, Alexander und LEMPITSKY, Victor S.: Neural Codes for Image Retrieval. *CoRR* (2014), Bd. abs/1404.1777, URL http://arxiv.org/abs/1404.1777

[Bao16] BAO, Yu; LI, Haojie; FAN, Xin; LIU, Risheng und JIA, Qi: Region-based CNN for Logo Detection, in: *Proceedings of the International Conference on Internet Multimedia Computing and Service*, ICIMCS'16, ACM, New York, NY, USA, S. 319–322, URL http://doi.acm.org/10.1145/3007669.3007728

[Ben09] BENGIO, Yoshua; LOURADOUR, Jérôme; COLLOBERT, Ronan und WESTON, Jason: Curriculum Learning (2009)

[Ben15] BENDALE, Abhijit und BOULT, Terrance E.: Towards Open Set Deep Networks. *CoRR* (2015), Bd. abs/1511.06233, URL http://arxiv.org/abs/1511.06233

[Bia15] BIANCO, Simone; BUZZELLI, Marco; MAZZINI, Davide und SCHETTINI, Raimondo: Logo recognition using cnn features, in: *International Conference on Image Analysis and Processing*, Springer, S. 438–448

[Bia17] BIANCO, Simone; BUZZELLI, Marco; MAZZINI, Davide und SCHETTINI, Raimondo: Deep learning for logo recognition. *Neurocomputing* (2017), Bd. 245: S. 23–30, URL http://www.sciencedirect.com/science/article/pii/S0925231217305660

[Cha14] CHATFIELD, K.; SIMONYAN, K.; VEDALDI, A. und ZISSERMAN, A.: Return of the Devil in the Details: Delving Deep into Convolutional Nets, in: *British Machine Vision Conference*

[Cor95] CORTES, Corinna und VAPNIK, Vladimir: Support-Vector Networks. *Mach. Learn.* (1995), Bd. 20(3): S. 273–297, URL http://dx.doi.org/10.1023/A:1022627411411

[Dai16] DAI, Jifeng; LI, Yi; HE, Kaiming und SUN, Jian: R-FCN: Object Detection via Region-based Fully Convolutional Networks. *CoRR* (2016), Bd. abs/1605.06409, URL http://arxiv.org/abs/1605.06409

[Dal05] DALAL, Navneet und TRIGGS, Bill: Histograms of Oriented Gradients for Human Detection, in: *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1 - Volume 01*, CVPR '05, IEEE Computer Society, Washington, DC, USA, S. 886–893, URL http://dx.doi.org/10.1109/CVPR.2005.177

[Den09] DENG, J.; DONG, W.; SOCHER, R.; LI, L.-J.; LI, K. und FEI-FEI, L.: ImageNet: A Large-Scale Hierarchical Image Database, in: *CVPR09*

[Don13] DONAHUE, Jeff; JIA, Yangqing; VINYALS, Oriol; HOFFMAN, Judy; ZHANG, Ning; TZENG, Eric und DARRELL, Trevor: DeCAF: A Deep Convolutional Activation Feature for Generic Visual Recognition. *CoRR* (2013), Bd. abs/1310.1531, URL http://arxiv.org/abs/1310.1531

[Egg15] EGGERT, Christian; WINSCHEL, Anton und LIENHART, Rainer: On the Benefit of Synthetic Data for Company Logo Detection, in: *Proceedings of the 23rd ACM International Conference on Multimedia*, MM '15, ACM, New York, NY, USA, S. 1283–1286, URL http://doi.acm.org/10.1145/2733373.2806407

[Eve07] EVERINGHAM, M.; VAN GOOL, L.; WILLIAMS, C. K. I.; WINN, J. und ZISSERMAN, A.: The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results, http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html (2007)

[Gir13] GIRSHICK, Ross B.; DONAHUE, Jeff; DARRELL, Trevor und MALIK, Jitendra: Rich feature hierarchies for accurate object detection and semantic segmentation. *CoRR* (2013), Bd. abs/1311.2524, URL http://arxiv.org/abs/1311.2524

[Gir16a] GIRSHICK, Ross: Fast R-CNN, http://www.robots.ox.ac.uk/~tvg/publications/talks/fast-rcnn-slides.pdf (2016)

[Gir16b] GIRSHICK, Ross; DONAHUE, Jeff; DARRELL, Trevor und MALIK, Jitendra: Region-Based Convolutional Networks for Accurate Object Detection and Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* (2016), Bd. 38(1): S. 142–158, URL http://dx.doi.org/10.1109/TPAMI.2015.2437384

[Gir17] GIRSHICK, Ross: py-faster-rcnn, https://github.com/rbgirshick/py-faster-rcnn (2017)

[Gor16] GORDO, Albert; ALMAZÁN, Jon; REVAUD, Jérôme und LARLUS, Diane: Deep Image Retrieval: Learning global representations for image search. *CoRR* (2016), Bd. abs/1604.01325, URL http://arxiv.org/abs/1604.01325

[Had06] HADSELL, Raia; CHOPRA, Sumit und LECUN, Yann: Dimensionality reduction by learning an invariant mapping, in: *In Proc. Computer Vision and Pattern Recognition Conference (CVPR?06*, IEEE Press

[Har88] HARRIS, Chris und STEPHENS, Mike: A combined corner and edge detector, in: *In Proc. of Fourth Alvey Vision Conference*, S. 147–151

[He14] HE, Kaiming; ZHANG, Xiangyu; REN, Shaoqing und SUN, Jian: Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. *CoRR* (2014), Bd. abs/1406.4729, URL http://arxiv.org/abs/1406.4729

[He15] HE, Kaiming; ZHANG, Xiangyu; REN, Shaoqing und SUN, Jian: Deep Residual Learning for Image Recognition. *CoRR* (2015), Bd. abs/1512.03385, URL http://arxiv.org/abs/1512.03385

[He17] HE, Kaiming; GKIOXARI, Georgia; DOLLÁR, Piotr und GIRSHICK, Ross: Mask R-CNN. *arXiv preprint arXiv:1703.06870* (2017)

[Her16a] HERRMANN, Christian; WILLERSINN, Dieter und BEYERER, Jürgen: Low-Quality Video Face Recognition with Deep Networks and Polygonal Chain Distance, in: *Proceedings of the 2016 International Conference on Digital Image Computing: Techniques and Applications (DICTA)*, IEEE, Gold Coast, Australia, S. 244–250

[Her16b] HERRMANN, Christian; WILLERSINN, Dieter und BEYERER, Jürgen: Low-Resolution Convolutional Neural Networks for Video Face Recognition, in: *Proceedings of the 13th IEEE International Conference on Advanced Video and Signal Based Surveillance, IEEE, 2016.*, IEEE, Colorado Springs, USA

[Hin12] HINTON, Geoffrey E.; SRIVASTAVA, Nitish; KRIZHEVSKY, Alex; SUTSKEVER, Ilya und SALAKHUTDINOV, Ruslan: Improving neural networks by preventing co-adaptation of feature detectors. *CoRR* (2012), Bd. abs/1207.0580, URL `http://dblp.uni-trier.de/db/journals/corr/corr1207.html#abs-1207-0580`

[Hoi15] HOI, Steven C. H.; WU, Xiongwei; LIU, Hantang; WU, Yue; WANG, Huiqiong; XUE, Hui und WU, Qiang: LOGO-Net: Large-scale Deep Logo Detection and Brand Recognition with Deep Region-based Convolutional Networks. *CoRR* (2015), Bd. abs/1511.02462, URL `http://arxiv.org/abs/1511.02462`

[Hu13] HU, Rui und COLLOMOSSE, John: A Performance Evaluation of Gradient Field HOG Descriptor for Sketch Based Image Retrieval. *Comput. Vis. Image Underst.* (2013), Bd. 117(7): S. 790–806, URL `http://dx.doi.org/10.1016/j.cviu.2013.02.005`

[Hua16] HUANG, Gao; LIU, Zhuang und WEINBERGER, Kilian Q.: Densely Connected Convolutional Networks. *CoRR* (2016), Bd. abs/1608.06993, URL `http://arxiv.org/abs/1608.06993`

[Ian15] IANDOLA, Forrest N.; SHEN, Anting; GAO, Peter und KEUTZER, Kurt: DeepLogo: Hitting Logo Recognition with the Deep Neural Network Hammer. *CoRR* (2015), Bd. abs/1510.02131, URL `http://arxiv.org/abs/1510.02131`

[ILS16] ImageNet Large Scale Vision Recognition Challenge - Ordered by classification error, `http://image-net.org/challenges/LSVRC/2016/results` (2016), accessed: 2017-06-11

[Iof15] IOFFE, Sergey und SZEGEDY, Christian: Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *CoRR* (2015), Bd. abs/1502.03167, URL `http://arxiv.org/abs/1502.03167`

[Jad14] JADERBERG, M.; SIMONYAN, K.; VEDALDI, A. und ZISSERMAN, A.: Reading Text in the Wild with Convolutional Neural Networks. *arXiv preprint arXiv:1412.1842* (2014)

[Jia14] JIA, Yangqing; SHELHAMER, Evan; DONAHUE, Jeff; KARAYEV, Sergey; LONG, Jonathan; GIRSHICK, Ross; GUADARRAMA, Sergio und DARRELL, Trevor: Caffe: Convolutional Architecture for Fast Feature Embedding, in: *Proceedings of the 22Nd ACM International Conference on Multimedia*, MM '14, ACM, New York, NY, USA, S. 675–678, URL `http://doi.acm.org/10.1145/2647868.2654889`

[Jol09] JOLY, Alexis und BUISSON, Olivier: Logo retrieval with a contrario visual query expansion, in: *MM '09: Proceedings of the seventeen ACM international conference on Multimedia*, S. 581–584

[Kal11] KALANTIDIS, Y.; PUEYO, LG.; TREVISIOL, M.; VAN ZWOL, R. und AVRITHIS, Y.: Scalable Triangulation-based Logo Recognition, in: *in Proceedings of ACM International Conference on Multimedia Retrieval (ICMR 2011)*, Trento, Italy

[Kri12] KRIZHEVSKY, Alex; SUTSKEVER, Ilya und HINTON, Geoffrey E: ImageNet Classification with Deep Convolutional Neural Networks, in: F. Pereira; C. J. C. Burges; L. Bottou und K. Q. Weinberger (Herausgeber) *Advances in Neural Information Processing Systems 25*, Curran Associates, Inc. (2012), S. 1097–1105, URL http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf

[LeC89] LECUN, Y.; BOSER, B.; DENKER, J. S.; HENDERSON, D.; HOWARD, R. E.; HUBBARD, W. und JACKEL, L. D.: Backpropagation Applied to Handwritten Zip Code Recognition. *Neural Comput.* (1989), Bd. 1(4): S. 541–551, URL http://dx.doi.org/10.1162/neco.1989.1.4.541

[Let12] LETESSIER, Pierre; BUISSON, Olivier und JOLY, Alexis: Scalable mining of small visual objects, in: *Proceedings of the 20th ACM international conference on Multimedia*, ACM, S. 599–608

[Liu15] LIU, Wei; ANGUELOV, Dragomir; ERHAN, Dumitru; SZEGEDY, Christian und REED, Scott E.: SSD: Single Shot MultiBox Detector. *CoRR* (2015), Bd. abs/1512.02325, URL http://dblp.uni-trier.de/db/journals/corr/corr1512.html#LiuAESR15

[Log17] Clearbit - Free logo API, https://clearbit.com/logo (2017), accessed: 2017-02-17

[Lon14] LONG, Jonathan; SHELHAMER, Evan und DARRELL, Trevor: Fully Convolutional Networks for Semantic Segmentation. *CoRR* (2014), Bd. abs/1411.4038, URL http://arxiv.org/abs/1411.4038

[Low04] LOWE, David G.: Distinctive Image Features from Scale-Invariant Keypoints. *Int. J. Comput. Vision* (2004), Bd. 60(2): S. 91–110, URL http://dx.doi.org/10.1023/B:VISI.0000029664.99615.94

[Mat92] MATAN, Ofer; BURGES, Christopher J.C.; CUN, Yann Le und DENKER, John S.: Multi-Digit Recognition Using A Space Displacement Neural Network, in: *Neural Information Processing Systems*, Morgan Kaufmann, S. 488–495

[Mil69] MILLER, Harold: The FROC Curve: a Representation of the Observer's Performance for the Method of Free Response. *The Journal of the Acoustical Society of America* (1969), Bd. 46(6(2)): S. 1473–1476

[Oli16] OLIVEIRA, Gonçalo; FRAZÃO, Xavier; PIMENTEL, André und RIBEIRO, Bernardete: Automatic Graphic Logo Detection via Fast Region-based Convolutional Networks. *CoRR* (2016), Bd. abs/1604.06083, URL http://arxiv.org/abs/1604.06083

[Pan14] PANDEY, Rohit; DI, Wei; JAGADEESH, Vignesh; PIRAMUTHU, Robinson und BHARDWAJ, Anurag: Cascaded sparse color-localized matching for logo retrieval, in: *2014 IEEE International Conference on Image Processing, ICIP 2014, Paris, France, October 27-30, 2014*, S. 2207–2211, URL https://doi.org/10.1109/ICIP.2014.7025447

[Qi17] QI, Cheng-Zuo; SHI, Cunzhao; WANG, Chunheng und XIAO, Baihua: Logo Retrieval Using Logo Proposals and Adaptive Weighted Pooling. *IEEE Signal Process. Lett.* (2017), Bd. 24(4): S. 442–445, URL https://doi.org/10.1109/LSP.2017.2673119

[Rad16] RADENOVIC, Filip; TOLIAS, Giorgos und CHUM, Ondrej: CNN Image Retrieval Learns from BoW: Unsupervised Fine-Tuning with Hard Examples. *CoRR* (2016), Bd. abs/1604.02426, URL http://arxiv.org/abs/1604.02426

[Red15] REDMON, Joseph; DIVVALA, Santosh Kumar; GIRSHICK, Ross B. und FARHADI, Ali: You Only Look Once: Unified, Real-Time Object Detection. *CoRR* (2015), Bd. abs/1506.02640, URL http://arxiv.org/abs/1506.02640

[Ren15] REN, Shaoqing; HE, Kaiming; GIRSHICK, Ross und SUN, Jian: Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks, in: C. Cortes; N. D. Lawrence; D. D. Lee; M. Sugiyama und R. Garnett (Herausgeber) *Advances in Neural Information Processing Systems 28*, Curran Associates, Inc. (2015), S. 91–99, URL http://papers.nips.cc/paper/5638-faster-r-cnn-towards-real-time-object-detection-with-region-proposal-netwo pdf

[Rom11a] ROMBERG, Stefan; PUEYO, Lluis Garcia; LIENHART, Rainer und VAN ZWOL, Roelof: Scalable Logo Recognition in Real-world Images, in: *Proceedings of the 1st ACM International Conference on Multimedia Retrieval*, ICMR '11, ACM, New York, NY, USA, S. 25:1–25:8, URL http://doi.acm.org/10.1145/1991996.1992021

[Rom11b] ROMBERG, Stefan; PUEYO, Lluis Garcia; LIENHART, Rainer und VAN ZWOL, Roelof: Scalable logo recognition in real-world images, in: *Proceedings of the 1st ACM International Conference on Multimedia Retrieval*, ICMR '11, ACM, New York, NY, USA, S. 25:1–25:8, URL http://www.multimedia-computing.de/flickrlogos/

[Rom13] ROMBERG, Stefan und LIENHART, Rainer: Bundle min-hashing for logo recognition., in: Ramesh Jain; Balakrishnan Prabhakaran; Marcel Worring; John R. Smith und Tat-Seng Chua (Herausgeber) *ICMR*, ACM, S. 113–120, URL http://dblp.uni-trier.de/db/conf/mir/icmr2013.html#RombergL13

[Ros58] ROSENBLATT, F.: The Perceptron: A Probabilistic Model for Information Storage and Organization in The Brain. *Psychological Review* (1958): S. 65–386

[Rus15] RUSSAKOVSKY, Olga; DENG, Jia; SU, Hao; KRAUSE, Jonathan; SATHEESH, Sanjeev; MA, Sean; HUANG, Zhiheng; KARPATHY, Andrej; KHOSLA, Aditya; BERNSTEIN, Michael; BERG, Alexander C. und FEI-FEI, Li: ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)* (2015), Bd. 115(3): S. 211–252

[Sch99] SCHAPIRE, Robert E.: A Brief Introduction to Boosting, in: *Proceedings of the 16th International Joint Conference on Artificial Intelligence - Volume 2*, IJCAI'99, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, S. 1401–1406, URL http://dl.acm.org/citation.cfm?id=1624312.1624417

[Ser13] SERMANET, Pierre; EIGEN, David; ZHANG, Xiang; MATHIEU, Michaël; FERGUS, Rob und LECUN, Yann: OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks. *CoRR* (2013), Bd. abs/1312.6229, URL http://dblp.uni-trier.de/db/journals/corr/corr1312.html#SermanetEZMFL13

[Sim13] SIMONYAN, K.; PARKHI, O. M.; VEDALDI, A. und ZISSERMAN, A.: Fisher Vector Faces in the Wild, in: *British Machine Vision Conference*

[Sim14] SIMONYAN, K. und ZISSERMAN, A.: Very Deep Convolutional Networks for Large-Scale Image Recognition. *CoRR* (2014), Bd. abs/1409.1556

[Siv03] SIVIC, Josef und ZISSERMAN, Andrew: Video Google: A Text Retrieval Approach to Object Matching in Videos, in: *Proceedings of the Ninth IEEE International Conference on Computer Vision - Volume 2*, ICCV '03, IEEE Computer Society, Washington, DC, USA, S. 1470–, URL http://dl.acm.org/citation.cfm?id=946247.946751

[Su16] SU, Hang; ZHU, Xiatian und GONG, Shaogang: Deep Learning Logo Detection with Data Expansion by Synthesising Context. *CoRR* (2016), Bd. abs/1612.09322, URL http://arxiv.org/abs/1612.09322

[Sze14] SZEGEDY, Christian; LIU, Wei; JIA, Yangqing; SERMANET, Pierre; REED, Scott E.; ANGUELOV, Dragomir; ERHAN, Dumitru; VANHOUCKE, Vincent und RABINOVICH, Andrew: Going Deeper with Convolutions. *CoRR* (2014), Bd. abs/1409.4842, URL http://arxiv.org/abs/1409.4842

[Tol15] TOLIAS, Giorgos; SICRE, Ronan und JÉGOU, Hervé: Particular object retrieval with integral max-pooling of CNN activations. *CoRR* (2015), Bd. abs/1511.05879, URL http://arxiv.org/abs/1511.05879

[Tur17] TURSUN, Osman; AKER, Cemal und KALKAN, Sinan: A Large-scale Dataset and Benchmark for Similar Trademark Retrieval. *CoRR* (2017), Bd. abs/1701.05766, URL http://arxiv.org/abs/1701.05766

[Uij13] UIJLINGS, J.R.R.; VAN DE SANDE, K.E.A.; GEVERS, T. und SMEULDERS, A.W.M.: Selective Search for Object Recognition. *International Journal of Computer Vision* (2013), URL http://www.huppelen.nl/publications/selectiveSearchDraft.pdf

[Vio04] VIOLA, Paul und JONES, Michael J.: Robust Real-Time Face Detection. *Int. J. Comput. Vision* (2004), Bd. 57(2): S. 137–154, URL http://dx.doi.org/10.1023/B:VISI.0000013087.49260.fb

[Xie16] XIE, Saining; GIRSHICK, Ross B.; DOLLÁR, Piotr; TU, Zhuowen und HE, Kaiming: Aggregated Residual Transformations for Deep Neural Networks. *CoRR* (2016), Bd. abs/1611.05431, URL http://arxiv.org/abs/1611.05431

[Yan16] YAN, Ke; WANG, Yaowei; LIANG, Dawei; HUANG, Tiejun und TIAN, Yonghong: CNN vs. SIFT for Image Retrieval: Alternative or Complementary?, in: *Proceedings of the 2016 ACM on Multimedia Conference*, MM '16, ACM, New York, NY, USA, S. 407–411, URL http://doi.acm.org/10.1145/2964284.2967252

[Zei13] ZEILER, Matthew D. und FERGUS, Rob: Visualizing and Understanding Convolutional Networks. *CoRR* (2013), Bd. abs/1311.2901, URL http://arxiv.org/abs/1311.2901

[Zit14] ZITNICK, Larry und DOLLAR, Piotr: Edge Boxes: Locating Object Proposals from Edges, in: *ECCV*, European Conference on Computer Vision, URL https://www.microsoft.com/en-us/research/publication/edge-boxes-locating-object-proposals-from-edges/

List of Figures

List of Tables

Glossary

BN - Batch Normalization

CL - Curriculum Learning

CNN - Convolutional Neural Network

DIR - Detection and Identification Rate

DR - Detection Rate

FC - Fully Connected Layer

FCN - Fully Convolutional Neural Network

FP - False Positive

FROC - Free-response Receiver Operating Characteristic

GT - Ground truth

HOG - Histogram of Oriented Gradients

HSV - Hue - Saturation - Value

IoU - Intersection over Union

ILSVRC - ImageNet Large Scale Visual Recognition Challenge

MAC - Maximum Activations of Convolutions

mAP - mean Average Precision

R-CNN - Region-Based Convolutional Neural Network

ReLU - Rectified Linear Unit

ResNet - Residual Network

ROC - Receiver Operating Characteristic

RoI - Region of Interest

RPN - Region Proposal Network

SIFT - Scale and Translation Invariant Features

SPP - Spatial Pooling Pyramid

SSD - Single Shot MultiBox Detector

SVM - Support Vector Machines

TP - True Positive

YOLO - You Only Look Once

Appendix

List of all brands with number of groundtruths RoIs

Brand	Number	Brand	Number	Brand	Number
Acerbis	127	Carglass	18	Fideconto	22
Adidas	649	Carlsberg	706	Fifa	4
Adidas-text	60	Cercol	30	Fila	59
AEG	5	Cerutticaffe	85	Fischer	3
Afxgroup	4	Chanel	81	Fmfiduciaria	27
AIL	33	Chimay	605	FMV	3
Airness	11	Citroen	129	Fontanaprint	10
Albatros	29	Citroen-text	197	Ford	351
Aldi	332	Cittadilugano	3	Fornoni	29
Allianz	73	Cocacola	767	Fosters	372
Amag	7	Cofidis	23	Garagestadio	8
Apple	662	Conforama	14	GC	1
Ascensorifalconi	29	Corona	739	Gdrive	1
Assijuris	2	Corriere	9	Generali	28
Assimedia	4	Corti	30	Google	259
Audi	116	Delfino	1	Grafica	8
Audi_text	57	Dexia	233	Grano	2
Bancastato	29	DHL	810	GTL	29
Base	161	Dicasterosport	45	GTR	10
Bauer	64	DNB	32	Gucci	80
Bauhaus	3	Donada	22	Guiness	466
Bayard	6	Eleclerc	14	Head	13
BCVSWKB	8	Emozioni	27	Heineken	388
Becks	300	Engelbertstrauss	43	Helvetia	14
Belarusbank	21	Erdinger	629	HH	82
Bfgoodrich	86	Errea	11	Honda	46
BIK	64	Esso	303	HP	479
Bits	34	Europadotti	22	HRS	2
BKW	10	Eventmore	48	Ilmassimo	5
BMW	674	Fclugano	33	Implenia	79
Bouigues	14	Fcluganocom	13	Infiniti	10
Bredobau	35	Fcluganologo	26	Insegne	7
Bridgestone	31	Fedex	647	Intel	38
Bridgestone-text	63	Ferrari	724	Jackwolfskin	30

Jacuzzi	42	Radio3i	13	Superleague_text	27
Kalma	1	Raiffeisen	274	Supreme	75
Kappa	9	Raiffeisen_black	12	Swissairlines	661
Kataltherm	20	Raiffeisen_blue	26	Swisscom	69
Kia	136	Ranzisa	24	Swisslos	13
Lacoste	90	Rauch	41	Tamborini	37
Latettoia	20	Redbull	41	Teleclub	66
LF	2	Reebok	18	Telenor	7
Longines	14	Renzetti	4	Texaco	479
Longines_text	13	Reusch	5	Ticinonews	24
Marina	34	Rittersport	458	Tissot	940
Mastai	72	Roche	2	Titraduce	13
Mcdonalds	34	Schindler	8	TL	3
Mediati	30	Schoeffel	4	TNT	101
Mercedes	84	Securiton	51	Total	78
Milka	705	SFLCH	3	TPL	2
Mini	30	SGKB	1	Tsingtao	395
MK	78	Shell	429	Tstorandserviceag	1
NBC	30	Singha	280	Umbro	151
Nexus	6	Sion	1	Unicef	50
Nike	407	Skoda	91	Uniqua	1
Nissan	82	SNCF	7	Updatefitness	4
Nukyposa	35	Sparco	30	UPS	406
Nvidia	292	Spinelli	27	US_president	14
Ochsnersport	57	Sponser	26	Uvex	9
Orange	1	Sporttip	343	Veolia	12
Parcomarani	13	Sportxx	13	Visana	102
Paulaner	579	Sprite	64	Vodafone	100
Pepsi	791	SRF	950	VRT	10
Peugeot	6	Srg_ssr	13	VW	54
Phenix	4	Srgssr	80	Welovefootball	1
Porsche	35	Stadionwankdorf	14	Wuerth	2
Postfinance	587	Standard_liege	95	Yahoo	38
Prada	84	Starbucks	679	YB	31
Principeviaggi	10	Stellaartois	547	You	24
Progel	45	Stierlin	74	Zepter	2
Puma	292	Stoeckli	2	Zurich	817
Puma-text	27	Studiobimage	34		
Quick	56	Superleague	86		