Open Logo Detection Challenge

Hang Su[1]
http://www.eecs.qmul.ac.uk/~hs308/
Xiatian Zhu[2]
eddy@visionsemantics.com
Shaogang Gong[1]
https://www.eecs.qmul.ac.uk/~sgg/

[1] School of EECS
Queen Mary University of London
London E1 4NS, UK

[2] Vision Semantics Limited
327 Mile End Road
London E1 4NS, UK

Abstract

Existing logo detection benchmarks consider artificial deployment scenarios by assuming that large training data with fine-grained bounding box annotations for each class are available for model training. Such assumptions are often invalid in realistic logo detection scenarios where new logo classes come progressively and require to be detected with little or none budget for exhaustively labelling fine-grained training data for every new class. Existing benchmarks are thus unable to evaluate the true performance of a logo detection method in realistic and open deployments. In this work, we introduce a more realistic and challenging logo detection setting, called *Open Logo Detection*. Specifically, this new setting assumes fine-grained labelling only on a small proportion of logo classes whilst the remaining classes have no labelled training data to simulate the open deployment. We further create an open logo detection benchmark, called *QMUL-OpenLogo*, to promote the investigation of this new challenge. QMUL-OpenLogo contains 27,083 images from 352 logo classes, built by aggregating/refining 7 existing datasets and establishing an open logo detection evaluation protocol. To address this challenge, we propose a *Context Adversarial Learning* (CAL) approach to synthesising training data with coherent logo instance appearance against diverse background context for enabling more effective optimisation of contemporary deep learning detection models. Experiments show the performance advantage of CAL over existing state-of-the-art alternative methods on the more realistic and challenging open logo benchmark. The QMUL-OpenLogo benchmark is publicly available at https://qmul-openlogo.github.io/.

1 Introduction

Logo detection in unconstrained scene images is crucial for a variety of real-world vision applications, such as brand trend prediction for commercial research and vehicle logo recognition for intelligent transportation [24, 32, 33]. It is inherently a challenging task due to the presence of varying sized logo instances in arbitrary scenes with uncontrolled illumination, multi-resolution, occlusion and background clutter (Fig. 1 (c)). Existing logo detection methods typically consider a small number of logo classes with the need for large sized training data annotated with object bounding boxes [2, 14, 15, 19, 20, 24, 31, 32, 33]. Whilst this controlled setting allows for a straightforward adoption of the state-of-the-art general

object detection models [6, 28, 30], it is unscalable to dynamic real-world logo detection applications where more new logo classes become of interest during model deployment, with the availability of only their clean design images (Fig. 1(a)). To satisfy such incremental demands, prior methods are significantly limited by the extremely high cost needed for labelling a large set of per-class training logo images [34]. Whilst this requirement is significant for practical deployments, it is ignored in existing logo detection benchmarks which consider only the unscalable fully supervised learning evaluations.

This work considers the realistic and challenging open-ended logo detection challenge. To that end, we introduce a new **Open Logo Detection** problem, where we have limited fine-grained object bounding box annotation in real scene images for only a small proportion of logo classes (supervised) with the remaining classes (the majority) totally unlabelled (unsupervised). As logo is a visual symbol, we have the clean logo designs for all target classes (Fig. 1(a)). The *objective* is to establish a logo detection model for all logo classes by exploiting the small labelled training set and logo design images in a scalable manner. One approach to open logo detection is by jointly learning logo detection and classification as YOLO9000 [28] so that the model can learn to detect logo objects from the labelled training images while learn to classify all logo design images. This method relies on robust object detection generalisation learned from labelled classes to unlabelled classes, and rich appearance diversity of object instances. Both assumption are invalid in our setting. Another alternative approach is synthesising training data [37] which overlays logo designs with geometry/illumination variations into context images at random scales and locations. But, it introduces *appearance inconsistency* between logo instances and scene context (Fig. 3(a)), which may impede the model generalisation.

In this work, we address the open logo detection challenge by presenting a Context Adversarial Learning (CAL) approach to automatically generate context consistent synthetic training data. Specifically, the CAL takes as input artificial images with superimposed logo designs [37], and outputs corresponding images with context consistent logo instances. This is a pixel prediction process, which we formulate as an image-to-image translation problem in the Generative Adversarial Network framework [8].

Our contributions are: **(1)** We scale up logo detection to dynamic real-world applications without fine-grained labelled training data for newly coming logo classes and present a novel *Open Logo Detection* setting. This differs significantly from existing fully supervised logo detection problems with the exhaustive need for object instance box labelling for all classes and hence having poor deployment scalability in reality. To our knowledge, this is the first attempt of investigating such a scalable logo detection scenario in the literature. **(2)** We introduce a new QMUL-OpenLogo benchmark for providing a standard test of open logo detection and facilitating a like-for-like comparative evaluation in the future studies. QMUL-OpenLogo is created based on 7 publicly available logo detection datasets through a careful logo classes merging and filtering process along with a benchmarking evaluation protocol. **(3)** We propose a Context Adversarial Learning (CAL) approach to synthesising context coherent training data for enabling effective learning of state-of-the-art object detection models in order to tackle the open logo detection challenge in a scalable manner. Importantly, CAL requires no exhaustive human labelling therefore generally applicable to any unsupervised logo classes. Experiments show the performance advantage of CAL for open logo detection on the QMUL-OpenLogo benchmark in comparison to state-of-the-art approaches YOLO9000 [28] and Synthetic Context Logo (SCL) [37] in contemporary object detection frameworks.

2 Related Works

Logo Detection Traditional methods for logo detection rely on hand-crafted features and sliding window based localisation [2, 15, 19, 31, 32]. Recently, deep learning methods [11, 12, 20, 36, 37] have been proposed which use generic object detection models [6, 7, 28, 30]. However, these methods are not scalable to realistic large deployments due to the need for: (1) Accurately labelled training data per logo class; (2) Strong object-level bounding box annotations. One exception is [36, 38] where noisy web logo images are exploited without manual labelling of object instance boxes. This method exploits a huge quantity of data to mine sufficient correct logo images, and is restricted for non-popular and new brand logos which may lack web data. Moreover, all the above-mentioned methods assume the availability of real training images for ensuring model generalisation. This further reduces their scalability and usability in real-world scenarios when many logo classes have no training images from real scenes such as those newly introduced logos. In this work, we investigate this under-studied *Open Logo Detection* setting, where the majority of logo classes have no training data.

Synthetic Data There are previous attempts to exploit synthetic data for training deep CNN models. Peng et al. [26] used 3D CAD object models to generate 2D images by varying the projections and orientations to augment the training data in few-shot learning scenarios. This method is based on the R-CNN model [7] with the proposal generation component independent from fine-tuning the classifier, making the correlation between objects and background context suboptimal. The work of [40] used synthetic data rendered from 3D models against varying background to enhance the training images of a pose model. Su et al. [37] similarly generated synthetic images by overlaying logo instances with appearance changes on random background images. Rather than randomly placing exemplar objects [26, 37, 40], Georgakis et al. [5] performed object-scene compositing based on accurate scene segmentation, similar as [10] for text localisation. These existing works mostly aim to generate images with varying object appearance. In contrast, we consider the consistency between objects and the surrounding context for generating appearance coherent synthetic images. Conceptually, our method is complementary to the aforementioned approaches when applied concurrently.

3 QMUL-OpenLogo: Open Logo Detection Benchmark

For enabling open logo detection performance test, we need to establish a corresponding benchmark which the literature lacks. To that end, it is necessary to collect a large number of logo classes for simulating the real-world deployments at scales. Given the tedious process of logo class selection, image data collection and filtering, as well as fine-grained bounding box annotation [39], we propose to re-exploit the existing logo detection datasets.

Source data selection To maximise the context richness of logo images, we assemble 7 existing publicly accessible logo detection datasets (Table 1) sourced from diverse domains to establish the QMUL-OpenLogo evaluation benchmark. All these datasets together present significant logo variations and therefore represent the truthful logo detection challenge as encountered in real-world unconstrained deployments. We only used the test data of WebLogo-2M [36] since its training data are noisy without labelled object bounding boxes which are required for model performance evaluation.

Logo annotation and class refinement We need to make logo class definition consistent

Table 1: Statistics of logo detection datasets used for constructing the QMUL-OpenLogo benchmark. Scale is the ratio of the logo instance area to the whole image area.

Dataset	Logos	Images	min∼max (mean) Instances / Class	min∼max (mean) Scale (%)
FlickrLogos-27 [15]	27	810	35∼213 (80.52)	0.0160∼100.0 (19.56)
FlickrLogos-32 [33]	32	2,240	73∼204 (106.38)	0.0200∼99.09 (9.16)
Logo32plus [1]	32	7,830	132∼576 (338.06)	0.0190∼100.0 (4.51)
BelgaLogos [14]	37	1,321	2∼223 (57.08)	0.0230∼69.04 (0.91)
WebLogo-2M(Test) [36]	194	4,318	18∼204 (40.63)	0.0180∼99.67 (7.69)
Logo-In-The-Wild [11]	1196	9,393	1∼1080 (23.49)	0.0007∼95.91 (1.80)
SportsLogo [20]	20	1,978	108∼292 (152.25)	0.0100∼99.41 (9.89)
QMUL-OpenLogo	352	27,083	10∼1,902 (88.25)	0.0014∼100.0 (6.09)

Table 2: Train/val/test data split in the QMUL-OpenLogo benchmark.

Type	Classes	Train Img	Val Img	Test Img	Logo Design Img
Supervised	32	1,280	1,019	9,168	32 (1 per class)
Unsupervised	320	0	1,562	14,054	320 (1 per class)
Total	352	1,280	2,581	23,222	352 (1 per class)

in QMUL-OpenLogo provided that different definitions exist across datasets. In particular, `Logo-In-The-Wild` (LITW) treats different logo variations of the same brand as distinct logo classes. For example, Adidas trefoil/text are treated as two different classes in LITW but as one class in all other datasets. We adopted the latter more common criterion by merging all fine-grained same-brand logo classes from LITW. We combined all logo image data of the same logo class from all selected datasets. We also cleaned up erroneous annotations by removing those with the size of bounding box exceeds the whole image size and/or obviously wrong box coordinates (e.g. $x_{\min} > x_{\max}$). To ensure that each logo class has sufficient test data, we further removed those extremely small classes with less than 10 images. Moreover, we manually verified 1∼3 random images per class and filtered out those classes with incorrect labels on selected images. These refinements result in a QMUL-OpenLogo dataset with 27,189 images of 309 logo classes (Table 1).

Train/val/test data partition For model training and evaluation on the QMUL-OpenLogo dataset as a test benchmark, we standardise the train/val/test sets in the following two steps: (1) We split all logo classes into two disjoint groups: one is *supervised* with labelled bounding boxes in real images, and the other is *unsupervised*, i.e. the open logo detection setting. In particular, we selected all 32 classes in the popular FlickrLogo32 dataset [33] as supervised ones whilst the remaining 277 classes as unsupervised. (2) For each supervised class, we assigned the original `trainval` set (40 images per class) of FlickrLogo32 as the `train` data. For open logo detection, no real training images are available for unsupervised classes. Among the remaining images, we further performed a random 10%/90% split for the `val`/`test` partition on each class. The data split is summarised in Table 2.

Logo design images Similar to [37], we obtained the clean logo design images from the Google Image Search by querying the corresponding logo class name. These images define the logo detection tasks, one for each logo class (Fig. 1 (a)).

Benchmark properties The QMUL-OpenLogo benchmark has three characteristics: (1) Highly imbalanced logo frequency distribution ("Instances/Class" in Table 1); (2) Significant logo scale variation ("Scale" in Table 1); (3) Rich scene context (Fig. 1(c)). These factors are essential for creating a benchmark entailing the true performance test of logo detection algorithms in real-world large scale deployments.

Figure 1: Example images of the QMUL-OpenLogo benchmark. (a) Clean logo design images; (b) Training images; (c) Test images.

4 Synthesising Images by Context Adversarial Learning

In open logo detection, there are training images $\mathcal{D}^s$ only for a small number of supervised classes $\mathcal{C}^s$, whilst no training data for other unsupervised classes $\mathcal{C}^u$ (Table 2). To enable state-of-the-art detection models [22, 28, 30], we exploit the potential of synthetic training data. To this end, we propose a **Context Adversarial Learning** (CAL) approach for rendering logo instance against scene context and improving the appearance consistency.

4.1 Context Adversarial Learning

The proposed CAL takes as input initial synthetic images with logo objects to generate the corresponding *context consistent* synthetic images. These output images serve as additional training data for enhancing state-of-the-art detection model generalisation on real-world unconstrained images. CAL is therefore a *detection* data augmentation strategy with focus on logo context optimisation. Conceptually, it is totally complementary to other existing data augmentation methods widely adopted for training classification and detection deep learning models, e.g. flipping, rotation, random noise, scaling [17, 22, 28, 30]. In this study, we adopt the SCL [37] to generate the initial synthetic data, i.e. superimpose the logo design images with spatial and colour transformation into any given natural scene images (Fig. 3(a)).

Model Formulation We consider the CAL as an image-to-image "translation" problem, i.e. translating one representation x of a logo scene image into another y of the same content at the pixel level. We particularly focus on rendering the logo objects to be more consistent with the scene context. Recently, deep neural networks have been verified as strong models capable of learning to minimise a given objective loss function [9, 18]. A straightforward solution may be the common convolutional neural networks (CNNs) which can be supervised to minimise the Euclidean distance between the predicted and ground truth pixel values. However, such modelling may lead to blurring results provided that the objective loss is minimised by averaging all plausible outputs [13, 25]. How to generate realistic images, the core of CAL, remains a generally unsolved problem for CNNs.

Interestingly, this task is exactly the formulation purpose of the recently proposed Generative Adversarial Networks (GANs) [9, 8, 27, 35] – making the generated images indistinguishable from realistic ones. Unlike the manually designed loss functions in CNNs, a GAN model automatically learns a loss that tries to classify if an output image is real (e.g. context consistent) or fake (e.g. context inconsistent), while simultaneously training a generative model to minimise this loss. Blurry logo images are clearly fake and therefore well suppressed. Given the dependence on initial synthetic image in our context, we explore the image-conditioned GAN which learns a conditional generative model [8, 13]. Formally, the objective value function of a conditional GAN can be written as:

$$\mathcal{L}_{\text{cGAN}}(G,D) = \mathbf{E}_{x,y}[\log D(x,y)] + \mathbf{E}_{x,z}[\log\left(1 - D(x,G(x,z))\right)] \quad (1)$$

where the generator G tries to minimise this objective value against an adversarial discriminator D which instead tries to maximise the value. Inspired by the modelling benefits from combining the GAN objective with pixel distance loss [13, 25], we enhance the conditional adversarial loss (Eq. (1)) with an L_1 loss to further suppress the blurring possibility:

$$\mathcal{L}_{\text{cGAN}}(G,D) = \mathbf{E}_{x,y}[\log D(x,y)] + \mathbf{E}_{x,z}[\log\left(1 - D(x,G(x,z))\right)] + \lambda \mathbf{E}_{x,y,z}\|y - G(x,z)\|_1 \quad (2)$$

where λ controls the weight of the L_1 pixel matching loss. We empirically set $\lambda = 100$ in our experiments. As such, the generator learning is also tied with the task of being close to

the ground truth output y in addition to fooling the discriminator, whilst the discriminator learning remains unchanged. The optimal solution is:

$$G^* = \arg \min_G \max_D \mathcal{L}_{\text{cGAN}}(G, D) \tag{3}$$

The noise z input aims to learn mapping from a distribution (e.g. Gaussian [42]) to the target domain y. However, this strategy is often ineffective to capture the stochasticity of conditional distributions with the noise largely neglected [13, 23]. Fortunately, it is not highly necessary to fully model this distribution mapping in our problem due to the presence of potentially infinite synthetic images by SCL. More specifically, the variation of y can be more easily captured by sampling the input synthetic logo images, than modelling the entropy of the conditional distributions through learning a mapping from one distribution to another. Without z, the model still learns a mapping from x to y in a deterministic manner.

Network Architecture We adopt the same generator and discriminator architectures as [13]. Specifically, the generator is an encoder-decoder network in a U-Net architecture with the encoder being a 8-layer CNN net whilst the decoder with a minor structure. The discriminator is a 4-layers CNN net. All convolution layers use 4×4 filters with stride 2.

Training Images To train the CAL model, we need a set of training image pairs. For generalising the model to rendering the synthetic images by SCL [37] at test time, we also apply the SCL to automatically build the training data. Specifically, given any natural image $\boldsymbol{I}$, we select a region (either a random rectangle or object foreground) and render it by SCL transformation including image sharpening, median filtering, random colour changes and colour reduction. This results in a CAL training image pair $(\boldsymbol{I}, \boldsymbol{I}_{\text{SCL}})$ where $\boldsymbol{I}_{\text{SCL}}$ is the image with inconsistent object region. We select two image sources for enhancing context richness: (1) Non-logo background images from FlickrLogo32 [33] on which we use random rectangle regions to generate training pairs; and (2) MS COCO images [21] on which we utilise the object masks to make training pairs. Importantly, this method requires no additional labelling in creating training data. A number of examples are given in Fig. 2.

Figure 2: Example CAL training pairs generated based on (a) object masks of COCO images and (b) random regions of non-logo images.

Model Optimisation and Deployment Given the training data, we adopt the standard optimisation approach as [13] to train the CAL: we alternate between optimising the discriminator D, and then the generator G in each mini-batch SGD. We train G to minimise $\log D(x, G(x, z))$ rather than $\log(1 - D(x, G(x, z))$ as suggested in [8]. We slow down the learning of D by applying half gradient. To focus the CAL model on context inconsistent region, we additionally input the region mask together with $\boldsymbol{I}_{\text{SCL}}$ in another channel, leading to a 4-channels input. Once the CAL is trained, it can be used to perform context rendering on the SCL synthesised images. Example CAL synthesised images are shown in Fig. 3 (b).

Figure 3: Examples of synthetic logo images by (a) SCL [37] and (b) our CAL.

4.2 Multi-Class Logo Detection Model Training

Given both SCL and CAL synthesised images and real (supervised classes only) images, we train a pre-selected deep learning detection model [28, 30]. First training on synthetic data and then fine-tuning on real images may make the detection model biased towards supervised logo classes whilst significantly hurting the performance on other unsupervised classes, as we will show in the experiments (Table 4).

5 Experiments

Competitors We compared our CAL approach with two state-of-the-art approaches allowing for open logo detection: **(1)** SCL [37]: A state-of-the-art method for generating synthetic detection training data with random logo design transformation and unconstrained background context. This enables the exploitation of state-of-the-art detection models same as the CAL. We selected two strong deep learning detectors: Faster R-CNN [30] and YOLOv2 [28]. **(2)** YOLO9000 [28]: A state-of-the-art deep learning detection model based on the YOLO architecture [29]. This model is designed particularly to scale up the detector to a large number of classes without exhaustive object instance box labelling. The key idea is by jointly learning the model from both bounding box labelled training data of supervised classes and image level classification training data of all classes using mixture mini-batches. We adopt the softmax cross-entropy loss for classification rather than the hierarchy aware loss as used in [28], since there is no semantic hierarchy on logo classes. To improve the model classification robustness against uncontrolled context, we further performed context augmentation on logo design images (Fig. 1(a)) using the SCL method [37].

Performance Metric For the performance measure of logo detection, we adopted the standard Average Precision (AP) for each individual logo class, and the mean Average Precision (mAP) for all classes [4]. A detection is considered to be correct when the Intersection over Union (IoU) between the predicted and ground-truth exceeds 50%.

Implementation Details For CAL model optimisation, we adopted the Adam solver [16] at the learning rate of 0.0002 and momentum parameters $\beta_1 = 0.5$ and $\beta_2 = 0.999$. For SCL [37] and CAL, we generated 100 synthetic images per logo class (in total 35,200).

Comparative Evaluations The comparative results on the QMUL-OpenLogo benchmark are shown in Table 3. To look into the detailed performance, we further evaluate the performance on unsupervised and supervised logo classes, as well as big (46.7%) and small (53.3%) logo instances split with a threshold of 0.02 scale ratio. We have these observations: **(1)** All methods produce rather poor results ($< 14\%$ mAP) on the QMUL-OpenLogo benchmark, suggesting that the scalability of current solutions remains unsatisfied in open logo detection deployments with the need for further investigation. **(2)** YOLO9000 [28] yields

Table 3: Evaluation on the QMUL-OpenLogo benchmark. Uns Class: Unsupervised Class; Sup Class: Supervised Class. Metric: mAP (%). FR-CNN: Faster R-CNN. Abs/Rel Gain: the absolute/relative performance gain of CAL over SCL. **Red**: the best results.

Method	All Class	Uns Class	Sup Class	Big Logo	Small Logo
YOLO9000[28]	4.19	1.98	26.33	6.23	1.15
YOLOv2[28]+SCL[37]	12.10	8.75	45.58	17.66	5.92
YOLOv2[28]+**CAL**	**13.14**	**9.55**	**49.17**	**18.25**	**6.29**
Abs/Rel Gain (%)	1.04/8.60	0.80/9.14	**3.59**/7.88	0.59/3.34	0.37/**6.25**
FR-CNN[30]+SCL[37]	12.35	8.51	50.74	16.94	7.87
FR-CNN[30]+**CAL**	**13.13**	**9.34**	**51.03**	**17.68**	**8.69**
Abs/Rel Gain (%)	0.78/6.32	**0.83**/9.75	0.29/0.57	0.74/4.37	0.82/**10.41**

the weakest performance among all methods. This suggests that joint learning of object classification and detection in a single loss formulation is ineffective to solve this challenging problem, particularly when the classification training data (clean logo design images) are limited in appearance variations. It is also found that such modelling can negatively affect the performance on supervised classes with fine-grained labelled training data. **(3)** With CAL, YOLOv2 achieves the best logo detection performance. While the absolute gain of CAL over the state-of-the-art data synthesising method SCL is small, larger relative gains are achieved using either YOLOv2 or Faster R-CNN. **(4)** The accuracy on supervised logo classes is much better than that on unsupervised ones. This indicates the high reliance on the manually labelled training data for existing state-of-the-art detection models, and the unsolved challenge of learning from auto-generated synthetic images. **(5)** Small logos benefit the largest relative gain from CAL. This is reasonable because, given limited appearance details of small instances, the external contextual information becomes more important for achieving accurate localisation and recognition.

To further justify the weak performance of state-of-the-art methods on the new QMUL-OpenLogo challenge, we evaluated Faster R-CNN+CAL on the most popular benchmark FlickrLogos-32 [33]. We obtained 74.9% mAP, which closely matches the 73.3% mAP of [12] similarly using a deep CNN model.

Qualitative Examination Fig 4 shows four test examples by SCL and CAL based on Faster R-CNN. For big logo instances of "Danone" in clean background, both models succeed. For moderate "Chiquita" with viewpoint distortion and small "Fiat" with subtle appearance, the SCL model fails while CAL remains successful. For small "Kellogs" instance against complex background clutter, both model fail.

Further Analysis In Table 4, we evaluated the performance of Faster R-CNN (1) trained on the mixture of synthetic and real data, or (2) trained firstly on synthetic data and then fine-tuned real data. It is evident that the former produces better performance except on supervised classes. This is as expected since the latter will bias the model towards supervised logo classes in the fine-tuning stage whilst largely degrading the generalisation capability on unsupervised classes.

Fully Supervised Learning Evaluation To more extensively evaluate the QMUL-OpenLogo dataset, we further benchmarked a fully supervised learning setting where each logo class has real training data. In particular, for every logo class, we made a 60%/10%/30% train/val/test image split at random. The data statistics are detailed in Table 5. With Faster-RCNN, we

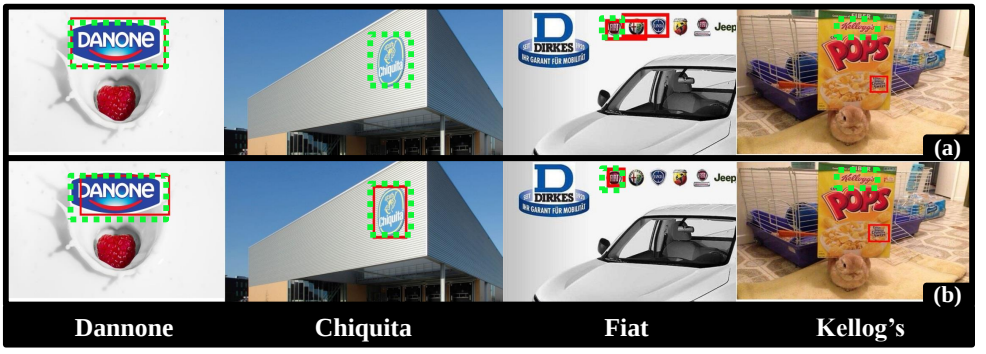

Figure 4: Example detection results by (a) SCL [37] and (b) CAL. Green dash boxes: ground-truth; Red solid boxes: model detection.

Table 4: Comparing the training methods of detection model (Faster R-CNN) using synthetic (syn) and real training data on the QMUL-OpenLogo benchmark. Uns Class: Unsupervised Class; Sup Class: Supervised Class. Metric: mAP (%).

Method	All Class	Uns Class	Sup Class	Big Logo	Small Logo
syn+real	**12.35**	**8.51**	50.74	**16.94**	**7.87**
syn→real	6.62	1.54	**57.39**	8.19	3.77

obtain a mAP performance of 48.3%, much higher than the best corresponding open logo detection rate 13.3% (Table 3).

Table 5: The data split statistics for the supervised learning setting on QMUL-OpenLogo.

Setting	Classes	Train	Val	Test	Total
Supervised Learning	352	15,975	2,777	8,331	27,083

6 Conclusion

In this work, we presented a new benchmark called QMUL-OpenLogo for enabling faithful performance test of logo detection algorithms in more realistic and challenging deployment scenarios. In contrast to existing closed benchmarks, QMUL-OpenLogo considers an open-end logo detection scenario where most classes are unsupervised – a simulation of incrementally arriving new logo classes without exhaustively labelled training data at fine-grained bounding box level during deployment. This benchmark therefore uniquely provides a more realistic evaluation of algorithms for logo detection in scalable and dynamic deployment with limited labelling budget. We further introduced a Context Adversarial Learning (CAL) approach to synthetic training data generation for enabling the learning optimisation of state-of-the-art supervised object detection model even given unsupervised logo classes. Empirical evaluations show the performance advantages of our CAL method over the state-of-the-art alternative detection and synthesising methods on the newly introduced QMUL-OpenLogo benchmark. We also provided detailed model performance analyses on different types of test data for giving insights on the specific challenges of the proposed more realistic open logo detection.

Acknowledgement

This work was partially supported by the China Scholarship Council, Vision Semantics Ltd, Royal Society Newton Advanced Fellowship Programme (NA150459), and Innovate UK Industrial Challenge Project on Developing and Commercialising Intelligent Video Analytics Solutions for Public Safety (98111-571149).

References

[1] Simone Bianco, Marco Buzzelli, Davide Mazzini, and Raimondo Schettini. Deep learning for logo recognition. *Neurocomputing*, 245:23–30, 2017.

[2] Raluca Boia, Alessandra Bandrabur, and Catalin Florea. Local description using multi-scale complete rank transform for improved logo recognition. In *IEEE International Conference on Communications*, pages 1–4, 2014.

[3] Emily L Denton, Soumith Chintala, Rob Fergus, et al. Deep generative image models using aï£ij laplacian pyramid of adversarial networks. In *Advances in Neural Information Processing Systems*, pages 1486–1494, 2015.

[4] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010.

[5] Georgios Georgakis, Arsalan Mousavian, Alexander C Berg, and Jana Kosecka. Synthesizing training data for object detection in indoor scenes. *arXiv preprint arXiv:1702.07836*, 2017.

[6] Ross Girshick. Fast r-cnn. In *IEEE International Conference on Computer Vision*, 2015.

[7] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

[8] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[9] Ian Goodfellow, Yoshua Bengio, Aaron Courville, and Yoshua Bengio. *Deep learning*, volume 1. MIT press Cambridge, 2016.

[10] Ankush Gupta, Andrea Vedaldi, and Andrew Zisserman. Synthetic data for text localisation in natural images. *arXiv e-prints*, 2016.

[11] Steven CH Hoi, Xiongwei Wu, Hantang Liu, Yue Wu, Huiqiong Wang, Hui Xue, and Qiang Wu. Logo-net: Large-scale deep logo detection and brand recognition with deep region-based convolutional networks. *arXiv preprint arXiv:1511.02462*, 2015.

[12] Forrest N Iandola, Anting Shen, Peter Gao, and Kurt Keutzer. Deeplogo: Hitting logo recognition with the deep neural network hammer. *arXiv*, 2015.

[13] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[14] Alexis Joly and Olivier Buisson. Logo retrieval with a contrario visual query expansion. In *ACM International Conference on Multimedia*, pages 581–584, 2009.

[15] Yannis Kalantidis, Lluis Garcia Pueyo, Michele Trevisiol, Roelof van Zwol, and Yannis Avrithis. Scalable triangulation-based logo recognition. In *ACM International Conference on Multimedia Retrieval*, page 20, 2011.

[16] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv*, 2014.

[17] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, pages 1097–1105, 2012.

[18] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553): 436, 2015.

[19] Kuo-Wei Li, Shu-Yuan Chen, Songzhi Su, Der-Jyh Duh, Hongbo Zhang, and Shaozi Li. Logo detection with extendibility and discrimination. *Multimedia tools and applications*, 72(2):1285–1310, 2014.

[20] Yuan Liao, Xiaoqing Lu, Chengcui Zhang, Yongtao Wang, and Zhi Tang. Mutual enhancement for detection of multiple logos in sports videos. In *IEEE International Conference on Computer Vision*, 2017.

[21] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*. 2014.

[22] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, and Scott Reed. Ssd: Single shot multibox detector. In *European Conference on Computer Vision*, 2016.

[23] Michael Mathieu, Camille Couprie, and Yann LeCun. Deep multi-scale video prediction beyond mean square error. *arXiv preprint arXiv:1511.05440*, 2015.

[24] Chun Pan, Zhiguo Yan, Xiaoming Xu, Mingxia Sun, Jie Shao, and Di Wu. Vehicle logo recognition based on deep learning architecture in video surveillance for intelligent traffic system. In *IET International Conference on Smart and Sustainable City*, pages 123–126, 2013.

[25] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2536–2544, 2016.

[26] Xingchao Peng, Baochen Sun, Karim Ali, and Kate Saenko. Learning deep object detectors from 3d models. In *IEEE International Conference on Computer Vision*, pages 1278–1286, 2015.

[27] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

[28] Joseph Redmon and Ali Farhadi. Yolo9000: better, faster, stronger. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[29] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[30] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems*, pages 91–99, 2015.

[31] Jerome Revaud, Matthijs Douze, and Cordelia Schmid. Correlation-based burstiness for logo retrieval. In *ACM International Conference on Multimedia*, pages 965–968, 2012.

[32] Stefan Romberg and Rainer Lienhart. Bundle min-hashing for logo recognition. In *Proceedings of the 3rd ACM conference on International conference on multimedia retrieval*, pages 113–120. ACM, 2013.

[33] Stefan Romberg, Lluis Garcia Pueyo, Rainer Lienhart, and Roelof Van Zwol. Scalable logo recognition in real-world images. In *Proceedings of the 1st ACM International Conference on Multimedia Retrieval*, page 25. ACM, 2011.

[34] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115 (3):211–252, 2015.

[35] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *Advances in Neural Information Processing Systems*, pages 2234–2242, 2016.

[36] Hang Su, Shaogang Gong, and Xiatian Zhu. Weblogo-2m: Scalable logo detection by deep learning from the web. In *Workshop of the IEEE International Conference on Computer Vision*, 2017.

[37] Hang Su, Xiatian Zhu, and Shaogang Gong. Deep learning logo detection with data expansion by synthesising context. *IEEE Winter Conference on Applications of Computer Vision*, 2017.

[38] Hang Su, Shaogang Gong, and Xiatian Zhu. Scalable deep learning logo detection. *arXiv preprint arXiv:1803.11417*, 2018.

[39] Hao Su, Jia Deng, and Li Fei-Fei. Crowdsourcing annotations for visual object detection. In *Workshops at the Twenty-Sixth AAAI Conference on Artificial Intelligence*, volume 1, 2012.

[40] Hao Su, Charles R Qi, Yangyan Li, and Leonidas J Guibas. Render for cnn: Viewpoint estimation in images using cnns trained with rendered 3d model views. In *IEEE International Conference on Computer Vision*, 2015.

[41] Andras Tüzkö, Christian Herrmann, Daniel Manger, and Jürgen Beyerer. Open set logo detection and retrieval. *arXiv preprint arXiv:1710.10891*, 2017.

[42] Xiaolong Wang and Abhinav Gupta. Generative image modeling using style and structure adversarial networks. In *European Conference on Computer Vision*, pages 318–335, 2016.