

面向 Logo 识别的合成数据生成方法研究

江玉朝, 吉立新, 高超, 李邵梅

(国家数字交换系统工程技术研究中心, 河南 郑州 450002)

摘要: 针对深度学习框架下 Logo 识别任务中可训练样本稀疏的问题, 提出了一种基于上下文的 Logo 数据合成算法, 该算法综合利用了 Logo 对象内部、Logo 周围邻域、Logo 与其他对象之间以及 Logo 所处场景等多种类型的上下文信息指导 Logo 图像的合成。在 FlickrLogos-32 数据集上的实验结果显示, 所提算法能够在不依赖额外手工标注的前提下, 提升 Logo 识别算法的性能 (mAP 提升 8.5%), 验证了该合成算法的有效性。

关键词: Logo 识别; 数据合成; 上下文; 深度学习; 数据增强

中图分类号: TP391

文献标识码: A

doi: 10.11959/j.issn.2096-109x.2018043

Research on synthesis data generation method for logo recognition

JIANG Yuchao, JI Lixin, GAO Chao, LI Shaomei

National Digital Switching System Engineering & Technological R&D Center, Zhengzhou 450002, China

Abstract: Aiming at the problem of training sample sparse in Logo recognition task under the deep learning framework, a Logo data synthesis algorithm based on contexts was proposed. The algorithm comprehensively utilizes various types of context information to guide the synthesis of Logo images, such as the interior of Logo object, the neighborhood of Logo object, the link between Logo object and other objects and the scene where Logo object lives in. The experimental results on the FlickrLogos-32 dataset show that the proposed algorithm can improve the performance of the Logo identification algorithm (mAP increase by 8.5%) without relying on additional manual annotation, verifying the effectiveness of the synthesis algorithm.

Key words: Logo recognition, data synthesis, context, deep learning, data augmentation

1 引言

Logo 识别是计算机视觉中一项具有挑战性的任务, 在很多领域有着广泛的应用, 如敏感视频识别^[1]、商标识别与产权保护^[2]、智能交通^[3]等。对于一般目标的识别, 深度学习方法已经取得了巨大的成功^[4-6]。通常, 构建一个目标识别的

深层神经网络模型需要大量手工标注的训练数据, 然而, 在 Logo 识别任务中能够获取到的公开数据集非常小, 现有的 Logo 数据集如表 1 所示^[7-12]。显然, 如此小的训练数据量对于学习具有数百万参数的深层模型是远远不够的, 通过增加手工标注来扩充数据集是解决这一问题较直接而简单的思路, 但是昂贵的标注成本以及大量的时间开销

收稿日期: 2018-03-24; 修回日期: 2018-05-02

通信作者: 江玉朝, jingjiujiang@qq.com

基金项目: 国家自然科学基金资助项目 (No.61601513)

Foundation Item: The National Natural Science Foundation of China (No.61601513)

往往令人难以承受,同时,相比通用物体,很多情况下很难大量获取到包含 Logo 的真实场景图像。面对种类庞杂的 Logo 对象,试图通过自动的方式搜集大规模训练数据也显得较为困难,诸如网络爬虫一类的 Web 数据挖掘方法虽然在一些目标识别任务中起到了一定的作用^[13,14],但由于典型的 Web 图像及其元数据中一般不包含目标定位的相关标注信息,采用此类方法难以获得 Logo 对象的精确标注。

表 1 现有的 Logo 识别数据集

数据集	Logo 类别数	对象总数	图像总数	是否公开
BelgaLogos ^[7]	37	2 695	1 951	是
FlickrLogos-27 ^[8]	27	4 671	1 080	是
FlickrLogos-32 ^[9]	32	3 404 ^{注1}	2 240	是
LOGO-NET ^[10]	160	130 608	73 414	否
Logos-32plus ^[11]	32	12 302	7 830	是
TopLogo10 ^[12]	10	863	700	是

合成数据生成是指在不依赖手工标注的前提下自动生成与真实数据近似的合成数据,从而扩展训练数据集的方法,当没有充足的训练数据可用于训练大型深层网络时,该方法是手工标注海量数据的有效替代方案。例如, Gupta 等^[15]和 Jaderberg 等^[16]通过人工合成自然场景文本数据训练文本识别模型; Georgakis 等^[17]通过分割出语义级别的“桌子”“柜台”等可能的支撑平面来指导合成图像中目标的放置,辅助室内场景中的目标检测; Eggert 等^[18]使用合成数据训练 SVM 分类器进行公司 Logo 检测; Su 等^[12]在文献[18]的基础上,考虑了合成 Logo 图像上下文的多样性,进行了利用大规模合成的 Logo 图像训练深层模型的首次尝试,较大地提升了检测模型应对复杂背景的顽健性。而缺陷方面,文献[12,18]使用的合成 Logo 图像的方法过于朴素,对于合成图像与真实场景图像的拟合考虑严重不足,仅通过将变换后的 Logo 模板放于任意场景图像中的随机位置完成 Logo 图像的合成。这种方式会使合成 Logo 图像的上下文真实性严重不足,导致模型过

多地学习到合成图像的细节,而无法很好地泛化到真实场景图像上,图 1 给出了文献[12,18]中合成图像的示例。



(a) 文献[12]的合成示例



(b) 文献[18]的合成示例

图 1 文献[12,18]中的 Logo 图像合成示例

为解决深度学习框架下 Logo 识别任务中标注数据缺乏的问题,本文提出了一种新的合成训练数据的算法,算法在充分挖掘和利用 Logo 图像上下文信息的基础上,通过合成尽可能拟合真实场景的 Logo 图像,在不增加额外标注成本的前提下,实现 Logo 识别算法性能的提升。虽然这项工作不是合成 Logo 图像的首次尝试^[12,18],但是本文改进了以往合成 Logo 图像的简单思路,充分利用了 Logo 对象内部、Logo 周围邻域、Logo 与其他对象之间以及 Logo 所处场景等多种类型的上下文信息,同时考虑 Logo 对象的自身特点,使借助自动合成的 Logo 图像参与训练能够产生更为显著的算法性能增益。实验方面,本文基于 Faster R-CNN^[19]目标检测框架在 Logo 识别的 benchmark 数据集 FlickrLogos-32^[9]上进行了详细的实验验证,取得了当前基于合成 Logo 图像辅助 Logo 识别任务的最好结果 (mAP 58.9% VS. 54.8%^[12]),充分验证了本文合成算法的有效性。此外,本文基于合成数据前提下探究的诸多训练策略与方法具备一定的借鉴和推广意义。

2 基于上下文的 Logo 图像合成算法

图 2 给出了本文基于合成数据进行 Logo 识别的整体算法框架,其中,生成合成图像作为本文的算法核心,主要包括 Logo 模板选取、背景图像

注1 已与文献[12]的作者取得联系,证实文献[12]中关于 FlickrLogos-32 数据集对象数量的统计有误,正确数量为 3 404。

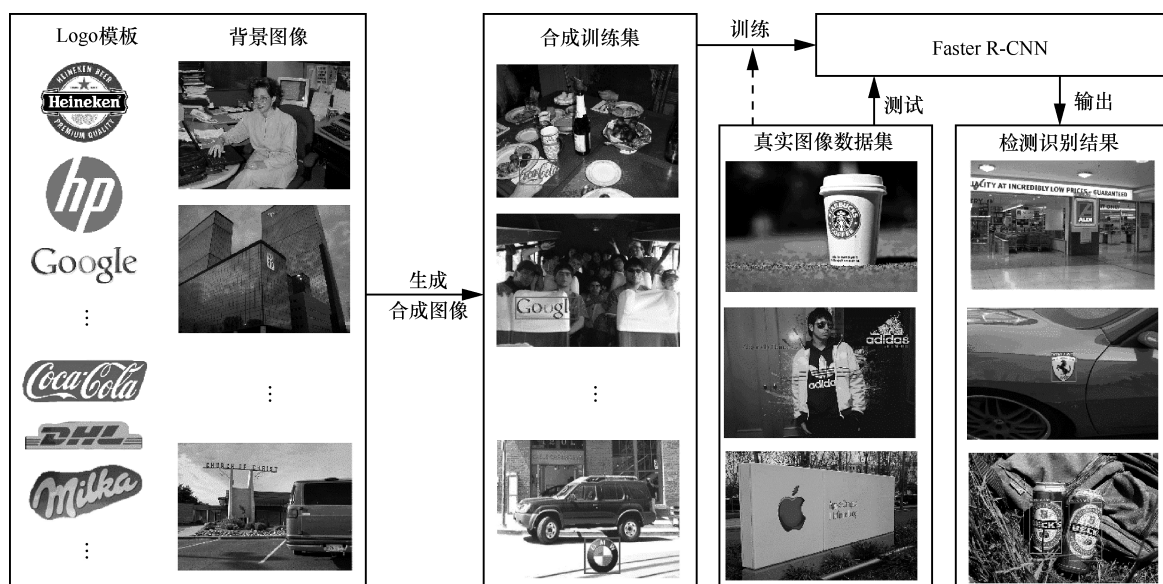


图2 本文基于合成数据的 Logo 识别整体框架

选取、Logo 模板变换以及 Logo 图像合成 4 个过程，下文将分别对其进行详细阐述。在 Logo 识别模型的训练方面，本文基本上沿用文献[12]中的顺序学习策略，该策略源自于课程学习^[20]中先易后难的学习思路，首先使用大规模的合成图像预训练深层模型，然后用少量的真实样本数据对模型进行精调。除此之外，本文通过实验探究发现，先用合成图像与真实图像的混合数据进行训练，再用真实图像进行精调，将取得更好的训练效果。

2.1 Logo 模板选取

为了进行 Logo 图像的合成，首先需要每类 Logo 用于合成的模板图像。文献[18]中使用像素级别的 Logo 掩码（如图 3(a)所示），此类模板的背景是不透明的，因此能够更多地保留 Logo 对象原生的细节，即 Logo 周围邻域的上下文信息；而缺陷也较为明显，一是像素级别的掩码需要通过烦琐的手工标注提取得到，二是合成图像中 Logo 邻域上下文信息的多样性有所欠缺。相反，文献[12]中使用背景完全透明的 Logo 模板（如图 3(b)所示），使任意合成图像中 Logo 邻域的像素完全由背景图像决定，而不是 Logo 模板，这样一来极大地提升了合成 Logo 邻域上下文的多样性，增强了识别模型应对复杂背景的顽健性；缺陷方面，一是由于采用随机放置 Logo 模板的做法，不可避免地会出现 Logo 与复杂背景混杂的情形（如图 3(b)所示），这显然与 Logo 本身的

设计原则^[21]相违背，导致合成图像与真实场景图像间的差异增大，二是由于过度地引入上下文信息，模型的误报率有所增加^[12]。



(a) 文献[18]选取的像素级别的 Logo 掩码



(b) 文献[12]选择的背景透明的 Logo 模板

图3 文献[12,18]中 Logo 模板选取对比

可见，以上 2 种 Logo 模板的选取做法是各有优劣而又相互补充的，考虑到后续合成数据集的构建与标注都是自动的，本文选择在 Logo 模板的预处理上花费一定的开销。因此，在 Logo 模板的选取上，本文综合选用了像素级别的 Logo 掩码以及背景完全透明的 Logo 图像（本文中两者合成的数量比为 1:1）。以 FlickrLogos-32 数据集的 32 类 Logo 为例，相应的 Logo 模板如图 4 所示。



图4 选取的 Logo 模板

2.2 背景图像选取

在现实场景中,目标一般不可能单独存在,它往往会与环境以及周围其他对象有着千丝万缕的联系,这就是通常所说的上下文信息^[22]。多种类型的上下文信息理论上已被证明在计算机视觉、图像处理领域扮演着非常重要的角色,能够提升检测识别算法的精度与速度^[23,24]。而作为承载合成 Logo 全局上下文信息的背景图像,文献[18]和文献[12]在其选择上仅考虑了上下文的多样性,使用 FlickrLogos-32 数据集中不含 Logo 的 6 000 张图片作为合成图像的背景来源,这样的简单处理必然会导致合成的 Logo 图像中蕴含很多不真实的上下文信息,进而影响练习得到的深层模型在真实场景中的泛化能力。如图 5 所示,Logo 十分生硬地出现在毫不相关的场景中,尽管这并不妨碍人类进行识别,但在实际训练过程中这些上下文信息完全不一致的合成样本很可能作为噪声数据存在。

因此,在背景图像选取方面,本文致力于减轻由于 Logo 模板与背景图像语义相关性太弱造成的上下文信息不一致的情况对算法性能的影响。具体来说,本文首先使用爬虫程序预先为每

类 Logo 在谷歌图片搜索网站上爬取了 300 张与之相关的场景图像,然后结合现今流行的基于 CNN 的场景分类模型 Places365-VGG 对爬取的场景图像进行批量分类,进而统计出每类 Logo 出现频率最高的 5 个 Top-1 场景作为合成图像的背景来源。其中,Places365-VGG 是文献[25]开源的针对大规模场景图像数据库 Places2 的子集 Places365 训练的 CNN 场景分类模型,网络结构采用的是 VGG-16,其在 Places365 的验证集和测试集上均取得了目前最高的 Top-1 分类精度。图 6 以 Starbucks 为例,给出了其背景图像选取的算法流程。

2.3 Logo 模板变换

传统的图像数据增强方法已被证明能够有效地丰富训练集、提升检测识别模型的顽健性和泛化能力^[26]。本文中不依赖传统数据增强方法对训练集进行扩充,而是采用图像合成作为训练数据的主要增强方法。实际上,在合成图像的基础上再进行数据增强的意义并不大,因此本文将数据增强应用于 Logo 模板的变换上。

由于实际自然场景中 Logo 呈现的尺度变化范围较大^[9],同时拍摄角度的不同可能导致 Logo



图5 不真实合成 Logo 图像示例

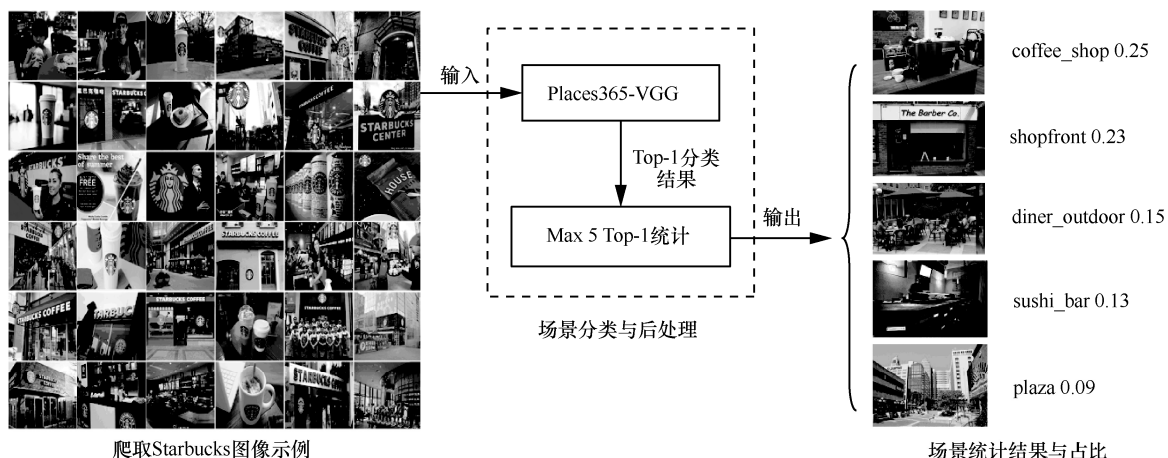


图6 背景图像选取算法流程

出现旋转、扭曲、变形以及部分遮挡等问题^[27]，此外，成像设备分辨率不同、光照条件变化等因素会显著增加 Logo 识别的难度。因此，为了尽可能拟合实际场景并且丰富合成图像中 Logo 的多样性，本文对 Logo 模板尝试进行了仿射变换、随机裁剪、颜色变换、高斯模糊等一系列增强变换。需要说明的是，每种变换之间相互独立且随机进行。下面以仿射变换为例给出其数学描述。由于卷积神经网络本身具有平移不变性，所以本文针对 Logo 模板并不进行平移变换，由此仿射变换的维度从 3 维降到 2 维，在 2 维平面上对于 Logo 模板 I 进行仿射变换的具体数学形式如下。

$$I^* = R_\theta P I$$

$$R_\theta = \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix} P = \begin{bmatrix} a & c \\ d & b \end{bmatrix}$$

其中，矩阵 R_θ 定义旋转变换，旋转角度 θ 本文选择在 $[-180^\circ, 180^\circ]$ 的范围内随机取值，但随机变换的概率控制在 0.1 以下，因为实际场景中 Logo 发生旋转的情况较少^[12]。矩阵 P 混合定义了尺度变换 $\text{Scale}(a, b)$ 和错切变换 $\text{Shear}(c, d)$ ，对于尺度变换，本文统计了 Flickrlogos-32 数据集整体对象的大小分布情况（如图 7 所示），选择控制 Logo 模板的长边变化为 40~250 pixel 之间的随机数，短边则进行等比例缩放；错切变换的参数选择为 $[0, 0.2]$ 之间的随机数。图 8 给出了本文 Logo 模板变换示意。

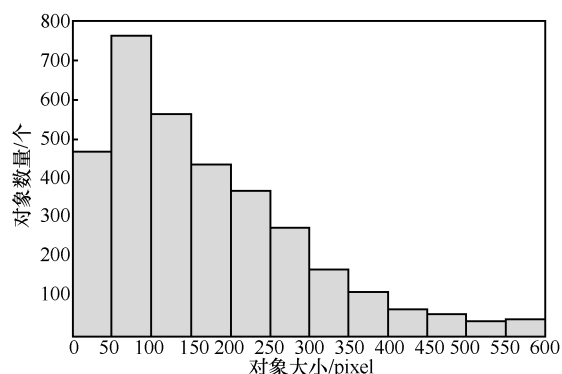


图7 FlickrLogos-32 数据集对象大小分布

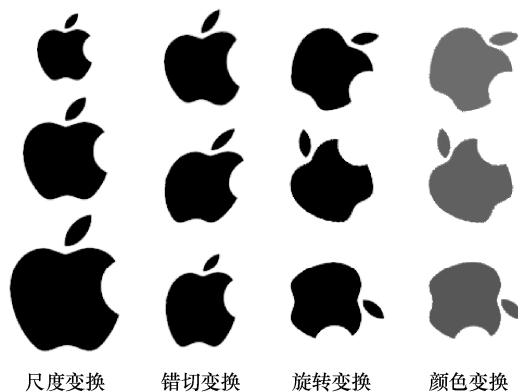


图8 本文 Logo 模板变换示意

2.4 Logo 图像合成

鉴于 Logo 对象具有平面性的特点，本文将随机变换后的 Logo 模板覆盖在语义层面上与其尽可能相关的随机背景图片上，从而实现 Logo 图像的合成。与文献[12,18]中将 Logo 模板随机放置的做法不同，本文特别关注 Logo 模板在背景中出现的位置。诚然，试图在背景中找到最“合理”的位

置涉及 Logo 对象与背景中其他对象之间复杂上下文关联关系的分析与处理,即图像的语义分割以及场景理解。而在本文的语境中,参与合成的对象较为特殊——Logo 作为独立对象进行合成,缺乏其“宿主”上下文信息的参与和指导,因而在语义层面上寻求合理位置的实现难度较高。

因此,现阶段本文的合成目标不追求能够找到语义层面上最“合理”的位置,而是尽可能寻求数据层面上较合适的位置。尽管有研究^[28]表明,视觉观察到的一致性效应很可能主要由场景的语义属性造成,而与一些低级视觉特征(如颜色、形状等)的关系不大。但依据 Logo 的设计原则^[21],Logo 对象确实应当出现在背景中纹理较简单、色彩较单调且与 Logo 本身颜色分布明显不同的区域。

具体而言,除了少数一些由纯文字符号组成的 Logo(如 Google),绝大多数 Logo 在设计时具备其主体色调^[21],同时在形状轮廓方面,Logo 一般都不呈现为规则的矩形结构(如图4所示)。因此在数据层面上,本文采用主颜色相似性比较算法,通过分别统计变换后的 Logo 模板与随机选取的背景放置区域在 HSV 颜色空间中各颜色出现的频率,选出最频繁出现的 2 种颜色作为各自的主色,如果两者的主色存在重叠,则更换背景图片而后重新随机选取背景放置区域,直到找到主色不重叠的位置进行合成。算法 1 给出了本文 Logo 图像合成算法流程。

算法 1 基于主颜色相似性比较的 Logo 图像合成算法

输入 变换后的 Logo 模板 I^* , 随机选取的背景图像 B

输出 合成的 Logo 图像 B^*

1) 在背景图像 B 中随机选取 Logo 模板 I^* 的放置区域 ROI, 并使 $Shape(I^*) = Shape(ROI)$ 。

2) 逐像素分别计算 I^* 与 ROI 在 HSV 颜色空间中的颜色分布。

3) 分别统计 $Max2Color(I^*)$, $Max2Color(ROI)$, 得到 I^* 和 ROI 各自的 2 个主色。 $Max2Color(\cdot)$ 返回类似[black, white]的二维数组。

4) if I^* 和 ROI 的主色存在重叠

5) 重新随机选取背景图像 B 。

6) return 1)

7) else

8) return $B^* = B.replaceWith(ROI, I^*)$

综合以上 4 个过程,本文合成算法不仅可以实现基于上下文的大规模 Logo 图像的自动合成,同时能够保证毫无遗漏地对合成图像中的每个 Logo 对象进行精确标注,图 9 给出了本文合成算法的最终效果示例。



图 9 本文合成算法的合成效果示例

3 实验与结果分析

3.1 数据集与评价指标

本文实验采用的数据集源自奥格斯堡大学多媒体计算和计算机视觉实验室维护并公开的 FlickrLogos-32 数据集,作为 Logo 识别的 benchmark 数据集,其通常用于评估多类 Logo 检测/识别以及真实图像上的 Logo 检索方法。FlickrLogos-32 共有 8 420 张图像(包含 6 000 张不含 Logo 的图像),分为 32 类 Logo,每类为 70 张且均具有较为平坦的表面。在实际训练过程中,本文严格按照该数据集的官方划分标准,划分每类 Logo 仅有 10 张图像作为训练样本,剩下的 60 张则作为测试样本,给定如此少的训练样本,一方面无疑对学习具有数百万参数的深层神经网络提出巨大挑战,另一方面也为验证训练样本缺乏条件下基于本文合成算法辅助 Logo 识别任务的有效性提供基础实验平台。

目标识别问题通常需要一定的评价指标来评估算法的性能,Logo 识别领域中常用的评价指标是 mAP (mean average precision)。mAP 综合表征了查准率 (precision) 和查全率 (recall),其值

越大，表明算法识别性能越好，本文中选择 mAP 进行算法性能的评估。

3.2 对比实验与分析

首先本文基本复现了文献[12]在 FlickrLogos-32 数据集上的实验结果，因为只有严格控制除了合成算法以外再无其他任何算法优化，本文合成算法的有效性验证及与文献[12]方法的对比才有意义。具体而言，本文使用 Faster R-CNN 目标检测算法作为 Logo 识别的算法框架，网络结构采用 VGG-16，并将其在 PASCAL VOC 2007 目标检测任务上预训练的模型作为网络的初始化权重，其他相关参数以及

训练策略均保持与文献[12]相同。在复现结果的基础上，利用本文合成算法自动地为每类 Logo 生成 100 张合成图像以及相应的标注数据，然后依次基于合成图像和真实图像进行 Logo 识别模型的训练。总体来说，与文献[12]的对比实验主要有以下 3 个过程。1) Reallmg：仅使用 320 张带标记的真实图像进行模型的训练。2) SynImg-32Cls：仅使用 3 200 张带标记的合成图像进行模型的训练。3) SynImg-32Cls+Reallmg：先使用 3 200 张合成图像进行模型的预训练，再使用 320 张真实图像对模型进行精调。相关实验结果如表 2 所示。需要说明的是，表 2 中

表 2 本文合成算法实验结果及与文献[12]的对比

方法	训练集/测试集划分 (每类图片数)	AP 值									mAP
		Adidas	Aldi	Apple	Becks	BMW	Carls	Chim	Coke		
		Corona	DHL	Erdi	Esso	Fedex	Ferra	Ford	Fost		
		Google	Guin	Hein	HP	Milka	Nvid	Paul	Pepsi		
		Ritt	Shell	Sing	Starb	Stel	Texa	Tsin	Ups		
RealImg ([12])	训练: 10 真实 测试: 60 真实	23.7	57.5	63.0	69.6	63.7	50.6	55.2	26.8	50.4%	
		79.0	25.8	61.2	44.2	45.9	80.6	64.3	43.2		
		47.7	58.2	61.8	21.3	19.4	17.4	48.2	17.8		
		34.8	45.8	71.8	70.2	79.6	56.7	56.9	52.2		
SynImg-32Cls ([12])	训练: 100 合成 测试: 60 真实	9.4	47.3	9.6	70.3	39.9	28.3	15.8	21.7	27.6%	
		6.1	11.1	4.1	44.7	22.9	60.9	43.6	28.8		
		23.0	16.7	43.1	9.9	4.6	1.1	39.1	9.7		
		22.7	38.3	15.5	65.6	28.7	55.1	27.4	20.1		
SynImg-32Cls + RealImg ([12])	训练: 100 合成 精调: 10 真实 测试: 60 真实	26.8	63.7	65.8	72.7	81.3	52.7	63.6	30.0	54.8%	
		76.0	31.5	63.0	52.2	54.3	90.0	84.0	46.6		
		58.0	52.6	65.2	23.2	24.0	12.5	54.1	23.6		
		37.9	45.6	75.0	73.8	79.0	64.2	57.4	54.4		
RealImg (Ours)	训练: 10 真实 测试: 60 真实	24.4	57.2	66.6	72.0	70.8	42.8	55.3	24.8	50.5%	
		82.8	29.5	62.5	44.1	42.7	87.2	59.3	39.9		
		51.2	54.6	65.1	24.2	15.5	16.9	52.3	17.4		
		32.3	44.7	72.7	69.7	77.3	62.5	52.6	44.3		
SynImg-32Cls (Ours)	训练: 100 合成 测试: 60 真实	38.6	54.0	34.6	44.1	17.6	18.4	33.9	20.6	32.6%	
		8.2	21.5	12.1	49.6	16.6	28.2	35.4	50.3		
		46.4	40.0	33.1	26.1	9.9	18.8	64.5	22.3		
		39.4	45.5	15.8	55.5	17.6	47.5	38.7	39.9		
SynImg-32Cls+ RealImg (Ours)	训练: 100 合成 精调: 10 真实 测试: 60 真实	31.0	69.0	74.3	78.0	76.6	51.1	62.1	29.8	58.5%	
		88.5	35.3	70.9	53.5	49.0	87.9	76.9	52.9		
		61.7	66.9	69.3	34.1	25.8	26.2	67.8	22.8		
		41.8	45.5	79.4	72.5	79.7	69.2	62.1	61.5		
SynImg-32Cls + RealImg (fusion) (Ours)	训练: 100 合成 +10 真实 (混合) 测试: 60 真实	36.2	61.3	30.1	46.9	25.3	30.8	32.2	22.3	34.2%	
		11.4	16.7	19.6	47.1	28.0	33.2	33.8	50.1		
		45.7	41.6	37.1	25.6	13.0	16.3	63.7	20.9		
		33.4	45.0	14.1	62.7	22.8	46.4	43.5	36.6		
SynImg-32Cls + RealImg (fusion) +RealImg (Ours)	训练: 100 合成 +10 真实 (混合) 精调: 10 真实 测试: 60 真实	31.1	71.2	71.6	71.7	84.4	48.0	66.3	28.5	58.9%	
		84.4	33.3	82.0	51.7	51.0	90.3	76.4	54.8		
		59.6	67.8	69.4	37.8	23.2	22.9	66.0	23.6		
		40.6	46.0	75.3	77.2	83.5	67.8	63.6	64.2		

第3大列实验数据分别是每类 Logo 的 AP (average precision) 值。

表2中, 单纯使用真实图像进行训练, 本文 (RealImg (Ours)) 取得了与 RealImg ([12]) 近似的实验结果 (50.5% VS 50.4%), 虽然 50.5% 是一次实验的结果, 但是实际上针对 RealImg 本文进行了多次实验, mAP 值基本在 50.0% 上下很小幅度变化, 导致实验结果不稳定的原因主要在于神经网络算法本身的随机性 (如 Faster R-CNN 使用随机梯度下降的优化算法), 因此本文基本复现了文献[12]在 FlickrLogos-32 数据集上的实验结果。而在此基础上, 采用本文合成算法的 SynImg-32Cls (Ours) 以及 SynImg-32Cls+RealImg (Ours) 均相对于文献[12]的方法取得明显优势 (32.6% VS 27.6%, 58.5% VS 54.8%), 充分验证了本文合成算法的有效性。更为值得一提的是, 取得如此性能的提升既不依赖于额外的手工标注, 也不需要像文献[12]那样构建庞大的 (463 类) 常见 Logo 模板集。

除此之外, 笔者还观察到以下情况。1) 仅使用 10 张真实图像训练, 针对 32 类 Logo 的识别 Faster R-CNN 就已经取得较好的训练结果 (mAP 50.5%), 这得益于迁移学习思想的运用, 网络的初始化权重本文使用的是在 PASCAL VOC 数据集上预训练的模型。2) 虽然 SynImg-32Cls (Ours) 相对于 SynImg-32Cls ([12]) 有很大的提升, 但是仅使用合成图像进行训练的效果仍与使用少量真实图像的方法存在很大的差距, 造成该现状的潜在原因可能是

真实图像与合成图像之间存在较大的分布差异, 模型在合成图像上学习到的细节难以泛化到真实图像中, 即通常所说的域漂移问题^[29]。从这个角度上看, 合成数据扩展训练集方法的关键在于如何优化合成算法以尽可能缩小合成图像与真实图像间的分布差异, 而本文算法在本质上也是通过利用多种类型的上下文信息在此方面进行尝试与努力。3) 基于先使用合成与真实的混合数据进行训练, 再用真实图像进行精调 (fusion + RealImg) 的训练策略, 模型将取得更好的训练结果 (mAP 58.9%)。文献[12]中为了验证课程学习思想的有效性, 仅限于使用混合数据进行训练, 没有再进一步地利用真实数据。实际上, 笔者认为“先混合再真实”的训练策略与课程学习“先易后难”的核心思想是一致的, 这也启发笔者扩展迁移学习的训练思路, 以往在训练一个深层神经网络之前, 笔者通常会在更通用的大数据集上进行预训练以获得网络的初始化权重, 而现在笔者可以预先将目标数据集与通用大数据集混合起来进行训练, 然后在目标数据集上进行精调, 这应当能取得可预期的更好结果。

图10给出了基于本文合成算法在 FlickrLogos-32 测试集上的部分实例测试结果, 可以发现算法对于目标的多尺度、多视角、旋转变形以及部分遮挡等情形具备一定的顽健性。

3.3 进一步的验证

为了定量地探究本文合成算法中每项改进细节对算法性能的影响, 本文针对算法中的每项改进进行了补充实验。这组实验的基线方法为表2



图10 基于本文算法在 FlickrLogos-32 测试集上的测试结果示例

中 SynImg-32Cls + RealImg (Ours)，每次实验控制忽略合成算法中的一项改进细节，训练策略均按照先使用 3 200 张合成图像进行模型的预训练，再使用 320 张真实图像对模型进行精调，具体有以下 5 个过程。1) Transparent Only: 用于合成的 Logo 模板仅选择背景完全透明的 Logo 图像。2) Pixel-level Only: 用于合成的 Logo 模板仅选择像素级别的 Logo 掩码。3) Random Context: 用于合成的背景图像任意选取，不考虑语义相关性。4) No Logo Transformations: 合成前不对 Logo 模板进行任何变换。5) Random Position: 合成图像中 Logo 模板的放置位置完全随机。相关实验结果如表 3 所示。

表 3 中，从宏观上可以看出，与基线方法相比，本文合成算法的每项改进对最终识别模型的性能提升都起到了促进作用，进一步验证了基于

本文合成算法的有效性。更具体地观察到如下事实。1) 采用像素级别的 Logo 掩码进行合成的效果要优于采用背景完全透明的情形 (58.1% VS 57.8%)，这表明 Logo 掩码中包含的局部上下文信息对于相关特征的学习是有所裨益的，但相较而言仍是综合使用两者的效果更好。2) 不考虑上下文一致性的随机背景选取做法会造成一定程度的性能损失 (56.7% VS 58.5%)，这表明合成图像中全局上下文信息对于特定对象的识别有着关键的辅助作用。3) 针对 Logo 模板进行的增强变换对于性能提升的效果最显著 (56.3% VS 58.5%)，这直接证明了传统数据增强对最终模型的识别性能以及泛化能力有着非常重要的影响。4) 仅在数据层面上优化合成位置的做法对算法性能会产生微小提升 (58.2% VS 58.5%)，性能提升受限的可能原因在于神经网络算法对于图像细节的感知方

表 3

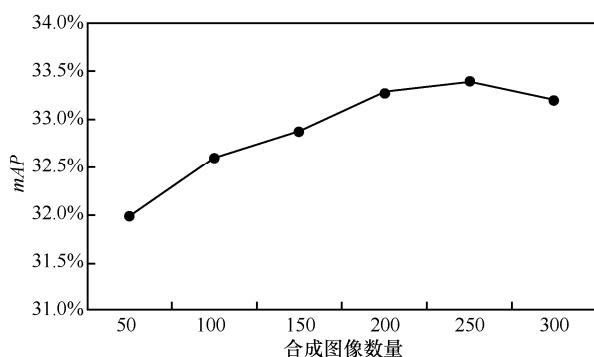
本文合成算法每项改进对性能影响的定量实验结果

方法	AP 值								mAP
	Adidas	Aldi	Apple	Becks	BMW	Carls	Chim	Coke	
	Corona	DHL	Erdi	Esso	Fedex	Ferra	Ford	Fost	
	Google	Guin	Hein	HP	Milka	Nvid	Paul	Pepsi	
	Ritt	Shell	Sing	Starb	Stel	Texa	Tsin	Ups	
SynImg-32Cls+ RealImg (Our Baseline)	31.0	69.0	74.3	78.0	76.6	51.1	62.1	29.8	58.5%
	88.5	35.3	70.9	53.5	49.0	87.9	76.9	52.9	
	61.7	66.9	69.3	34.1	25.8	26.2	67.8	22.8	
Transparent Only+ RealImg	41.8	45.5	79.4	72.5	79.7	69.2	62.1	61.5	57.8%
	32.8	69.7	75.2	77.6	76.8	51.3	61.8	29.1	
	88.0	35.9	71.9	52.9	43.9	87.5	76.3	47.8	
Pixel-level Only+ RealImg	59.3	61.5	68.3	33.2	24.8	24.3	67.4	18.5	58.1%
	42.6	47.5	78.6	73.6	80.3	69.2	61.3	61.2	
	32.7	66.7	73.8	75.5	77.4	48.3	59.7	32.0	
Random Context+ RealImg	87.5	37.5	71.7	53.6	52.2	87.7	72.2	53.5	56.7%
	63.8	65.9	71.3	29.7	26.8	25.4	62.9	25.4	
	36.5	48.6	75.4	75.3	79.3	66.6	63.4	59.3	
No Logo Transform+ RealImg	28.5	65.2	68.2	74.5	78.6	46.6	60.0	30.8	56.3%
	86.4	33.5	70.1	54.5	50.4	83.5	75.9	52.2	
	61.6	63.3	69.8	28.5	26.3	22.5	62.6	23.8	
Random Position+ RealImg	35.6	47.2	74.6	74.0	79.5	69.8	61.0	56.1	58.2%
	30.5	63.7	72.0	73.3	73.4	49.0	57.2	27.1	
	85.1	33.9	68.1	52.8	47.6	85.0	75.4	48.6	
	56.2	62.2	69.1	33.9	24.0	26.0	67.6	23.6	58.5%
	34.0	46.3	73.5	72.9	80.5	69.0	61.8	58.4	
	33.1	70.2	73.6	74.6	77.0	50.0	61.3	30.5	
	83.0	35.8	71.7	53.9	47.0	86.6	78.4	54.4	58.2%
	57.9	69.8	71.4	33.7	20.4	24.9	64.3	24.1	
	41.7	47.5	77.3	74.4	80.9	68.4	63.6	60.2	

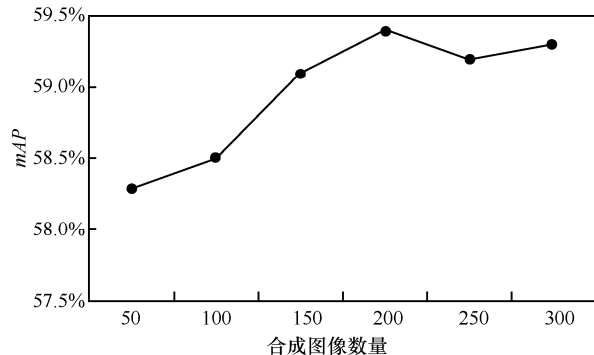
式与人类视觉系统存在一定的差异,合成图像中的一些低级视觉特征对于最终性能的影响较为有限。

最后,本文还探究了基于合成数据方法前提下参与训练的合成样本数量对算法性能的影响。

如图所示,图 11(a)为仅用合成图像进行训练的结果,图 11(b)为先用合成图像进行训练、再用真实图像进行精调的结果。从图中可以看出,适度增加参与训练的合成样本数量能够在一定程度上提升识别模型的性能。由于从该角度出发的探究更像是一项具体的调参工作,因此本文对此并未展开更加深入的研究。



(a) 仅用合成图像进行训练的结果



(b) 先用合成图像进行训练再用真实图像精调的结果

图 11 合成样本数量对算法性能的影响

4 结束语

本文针对深度学习框架下 Logo 识别任务中标注数据缺乏的问题,从自动合成大规模训练数据的角度,在现有合成思路的基础上,提出了一种基于上下文的 Logo 图像合成算法。通过在 FlickrLogos-32 数据集上的详细实验,表明当只有少量标注数据可用时,采用本文的合成算法能够在不依赖额外手工标注的前提下,实现 Logo 识别算法性能的较大幅度提升 (mAP 提升 8.5%,

50.4% VS 58.9%,如表 2 所示),充分验证了本文合成算法的有效性与优越性,同时进一步表明多种类型的上下文信息能够有效地提升目标识别算法的性能。此外,本文基于合成数据前提下探究的诸多训练策略与方法具备一定的借鉴和推广意义。由于对于合成图像中 Logo 位置的优化本文仍停留在数据层面,如何更深层次地利用上下文信息以及进一步缩小合成样本与真实样本间的分布差异将是本文下一步的研究方向。

参考文献:

- [1] 符亚彬. 基于 Logo 标志检测的暴恐视频识别系统的设计与实现[D]. 北京: 北京交通大学,2016.
FU Y B. Design and implementation of violence and fear video recognition system based on Logo mark detection[D]. Beijing: Beijing Jiaotong University,2016.
- [2] GAO Y, WANG F, LUAN H, et al. Brand data gathering from live social media streams[C]//ACM International Conference on Multimedia Retrieval. 2014:169.
- [3] PAN C, YAN Z, XU X, et al. Vehicle logo recognition based on deep learning architecture in video surveillance for intelligent traffic system[C]//IET International Conference on Smart and Sustainable City. 2013:123-126.
- [4] HE K, GKIOXARI G, DOLLAR P, et al. Mask R-CNN[C]//IEEE International Conference on Computer Vision. 2017:2980-2988.
- [5] WANG X, SHRIVASTAVA A, GUPTA A. A-Fast-RCNN: hard positive generation via adversary for object detection[C]//IEEE Conference on Computer Vision and Pattern Recognition. 2017: 3039-3048.
- [6] LIU W, ANGUELOV D, ERHAN D, et al. SSD: single shot MultiBox detector[M]//Computer Vision-ECCV 2016. Springer International Publishing, 2016:21-37.
- [7] JOLY A, BUISSON O. Logo retrieval with a contrario visual query expansion[C]//International Conference on Multimedia 2009. 2009:581-584.
- [8] KALANTIDIS Y, PUEYO L G, TREVISIOL M, et al. Scalable triangulation-based logo recognition[C]//ACM International Conference on Multimedia Retrieval. 2011:1-7.
- [9] ROMBERG S, PUEYO L G, LIENHART R, et al. Scalable logo recognition in real-world images[C]//ACM International Conference on Multimedia Retrieval. 2011:25.
- [10] HOI S C H, WU X, LIU H, et al. LOGO-Net: Large-scale deep logo detection and brand recognition with deep region-based convolutional networks[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2015, 46(5):2403-2412.
- [11] BIANCO S, BUZZELLI M, MAZZINI D, et al. Deep learning for logo recognition[J]. Neuro Computing, 2017, 245(C):23-30.
- [12] SU H, ZHU X, GONG S. Deep learning logo detection with data expansion by synthesising context[C]//IEEE Winter Conference on Applications of Computer Vision. 2017:530-539.
- [13] CHEN X, GUPTA A. Webly supervised learning of convolutional

- networks[C]//IEEE International Conference on Computer Vision. 2016:1431-1439.
- [14] SHRIVASTAVA A, GUPTA A, GIRSHICK R. Training region-based object detectors with online hard example mining[C]//IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2016). 2016:761-769.
- [15] GUPTA A, VEDALDI A, ZISSERMAN A. Synthetic data for text localisation in natural images[C]//IEEE Computer Vision and Pattern Recognition. 2016:2315-2324.
- [16] JADERBERG M, SIMONYAN K, VEDALDI A, et al. Reading text in the wild with convolutional neural networks[J]. International Journal of Computer Vision, 2016, 116(1):1-20.
- [17] GEORGAKIS G, MOUSAVIAN A, BERG A C, et al. Synthesizing training data for object detection in indoor scenes[C]//Robotics: Science and Systems. 2017.
- [18] EGGERT C, WINSCHER A, LIENHART R. On the benefit of synthetic data for company logo detection[C]//ACM International Conference on Multimedia. 2015:1283-1286.
- [19] REN S, HE K, GIRSHICK R, et al. Faster R-CNN: towards real-time object detection with region proposal networks[C]// International Conference on Neural Information Processing Systems. 2015:91-99.
- [20] BENGIO Y, COLLOBERT R, WESTON J. Curriculum learning[C]//ACM International Conference on Machine Learning. 2009:41-48.
- [21] LIU B. Modest proposal for the principle of logo design[J]. Packaging Engineering, 2005, 127(2):222-222.
- [22] OLIVA A, TORRALBA A. The role of context in object recognition[J]. Trends in Cognitive Sciences, 2007, 11(12):520.
- [23] MOTTAGHI R, CHEN X, LIU X, et al. The role of context for object detection and semantic segmentation in the wild[C]//IEEE Computer Vision and Pattern Recognition. 2014:891-898.
- [24] KATTI H, PEELEN M V, ARUN S P. How do targets, nontargets, and scene context influence real-world object detection?[J]. Attention Perception & Psychophysics, 2017(2):1-16.
- [25] ZHOU B, LAPEDRIZA A, KHOSLA A, et al. Places: a 10 million image database for scene recognition[J]. IEEE Trans Pattern Anal Mach Intell, 2017, 99: 1-1.
- [26] GUO J, GOULD S. Deep CNN ensemble with data augmentation for object detection[J]. Computer Science, 2015.
- [27] OLIVEIRA G, FRAZÃO X, PIMENTEL A, et al. Automatic graphic logo detection via fast region-based convolutional networks[C]//IEEE International Joint Conference on Neural Networks. 2016.
- [28] MUNNEKE J, BRENTARI V, PEELEN M. The influence of scene context on object recognition is independent of attentional focus[J]. Frontiers in Psychology, 2013, 4(8):552.
- [29] NGUYEN H V, HO H T, PATEL V M, et al. DASH-N: joint hierarchical domain adaptation and feature learning[J]. IEEE Transactions on Image Processing, 2015, 24(12):5479-5491.

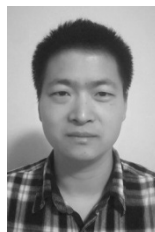
[作者简介]



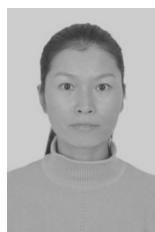
江玉朝（1994-），江苏盐城人，信息工程大学硕士生，主要研究方向为计算机视觉。



吉立新（1969-），江苏淮安人，博士，信息工程大学研究员，主要研究方向为通信与信息系统。



高超（1982-），河南郑州人，博士，信息工程大学助理研究员，主要研究方向为计算机视觉。



李邵梅（1982-），湖北钟祥人，博士，信息工程大学副研究员，主要研究方向为计算机视觉。