

基于深度学习航拍图像检测的梯度聚类算法

解博*, 朱斌, 张宏伟, 马旗, 张扬

国防科技大学脉冲功率激光技术国家重点实验室, 安徽 合肥 230037

摘要 针对在目标检测中现有方法检测速度慢的问题, 基于航拍图像中人造物体含有大量边缘的特点, 提出了一种基于梯度聚类的区域建议算法 (APM)。利用目标检测算法对提取的感兴趣区域进行检测, 在 DOTA (Dataset for Object deTecton in Aerial Images) 数据集上对算法的实时性和准确率进行了测试。研究表明, 所提算法极大地提升了目标检测算法对大尺寸、目标密集的航拍图像的检测速度, 该方法的召回率较高。

关键词 图像处理; 目标检测; 深度学习; 航拍图像; 梯度聚类; 区域建议方法

中图分类号 TP751.1

文献标识码 A

doi: 10.3788/LOP56.061007

Gradient Clustering Algorithm Based on Deep Learning Aerial Image Detection

Xie Bo*, Zhu Bin, Zhang Hongwei, Ma Qi, Zhang Yang

State Key Laboratory of Pulsed Power Laser Technology, National University of Defense Technology,
Hefei, Anhui 230037, China

Abstract An algorithm called gradient clustering based area proposal method (APM) is proposed to solve the problem that the existing methods are slow to detect objects, which is based on a large number of edges of artificial objects in aerial images. Then the extracted regions of interest are detected by the object detection method. The real-time performance and precision rate of this method are evaluated on the DOTA (Dataset for Object Detection in Aerial Images). The research results show that the proposed method greatly improves the detection speed of large-size, target-dense aerial images by the object detection algorithm, and still keeps a high recall rate.

Key words image processing; object detection; deep learning; aerial image; gradient clustering; area proposal method

OCIS codes 100.3008; 100.4996; 280.4991

1 引言

近年来, 由于遥感技术的发展^[1-2], 航拍图像的应用受到了广泛的关注。相比于地面拍摄图像, 航拍图像拥有更广阔的视野和更丰富的信息, 在侦查监视^[3]、城市规划、交通管制^[4]等领域有着广泛的应用, 这些应用普遍要求图像中的目标能够被准确检测。为了减小工作量, 实时地分析场景情况, 学者们提出了很多目标检测的算法^[5-16]。基于手工设计特征的目标检测算法^[5-9]只能检测特定目标, 而且准确率有限。基于深度学习的目标检测算法^[10-16]不仅在大规

模常规目标图像数据集竞赛中取得了好成绩, 如 Pascal VOC (The Pascal Visual Object Classes Challenge)^[17]、MSCOCO (Microsoft COCO; Common Objects in Context)^[18] 以及 ImageNet (A Large-Scale Hierarchical Image Database)^[19], 而且成功应用于实际应用中^[10]。因此, 基于深度学习的目标检测算法逐渐取代了基于手工设计特征的目标检测算法。

基于深度学习的目标检测算法解决了航拍图像中多类目标的检测问题, 并且提高了目标检测的准确率, 但是由于航拍图像的尺寸大, 目标检测的实时性问题一直没能被有效地解决。文献[11]在训练神经

收稿日期: 2018-08-17; 修回日期: 2018-09-20; 录用日期: 2018-10-12

基金项目: 国家自然科学基金(61271376)

* E-mail: bigboo@foxmail.com

网络时,采用选择性搜索算法对大尺寸航拍图像进行目标候选框提取,同时使用新的对目标旋转特性不敏感的目标函数,因为提取大量目标候选框的过程非常耗时,所以最后所达到的检测速度是每张图耗时8.77 s。文献[16]指出受目前图像处理器(GPU)的限制,深度卷积网络(CNN)不能处理大尺寸的图像。针对这个问题,Liu等^[16]利用滑窗法将大尺寸图像裁剪成多个小块后进行训练和检测,但是在一张图像上进行大量的目标检测严重影响了该算法检测速度。文献[17]对3类8种主要的OPM(Object Proposals Method)算法在航拍目标检测中的应用进行了研究,但并没有提出解决航拍图像检测速度慢的办法。

本文根据航拍图像中人造物体边缘轮廓明显且易聚集的特点,提出了一种基于图像梯度聚类的区域建议算法(APM),实现了对大尺寸航拍图像中感兴趣区域的快速提取。该算法根据人造物体轮廓明显的特点,利用Sobel梯度算子得到航拍图像的梯度图;利用STING(STatistical INformation Grid)聚类算法获得物体集中的聚类区域;在DOTA(Dataset for Object Detection in Aerial Images)^[20]数据集的基础上,结合SSD^[16](Single Shot MultiBox Detector)目标检测算法,对所提出的算法

的目标检测速度以及准确率进行了对比分析,并在理论上分析了图像大小、数据有效率等指标对该算法的实时性、召回率的影响,验证了该方案在提高航拍图像的目标检测速度上的有效性。

2 方法

2.1 SSD目标检测算法

2.1.1 SSD模型结构

SSD目标检测算法是由刘伟等提出的一种基于回归网络的目标检测模型,SSD的网络结构如图1所示。相比Faster R-CNN(Faster Region-Convolutional Neural Networks)目标检测模型,该算法没有生成区域建议的过程,因此极大地提高了检测速度。对于不同大小的目标,SSD算法利用不同卷积层的特征图进行融合达到目标检测的目的。SSD的主网络结构采用VGG16(Visual Geometry Group16)^[21],其主网络后增加了融合多层特征的融合层,最后输出用于分类的置信度以及位置回归用的位置偏移。目前SSD300算法(输入图像为固定尺寸:300 pixel×300 pixel)在Titan X显卡上的检测帧速度 V_0 为59 frame/s,在VOC2007测试集上的平均准确度(mAP)能达到74.3%,是最好的平衡速度与精度的算法之一。

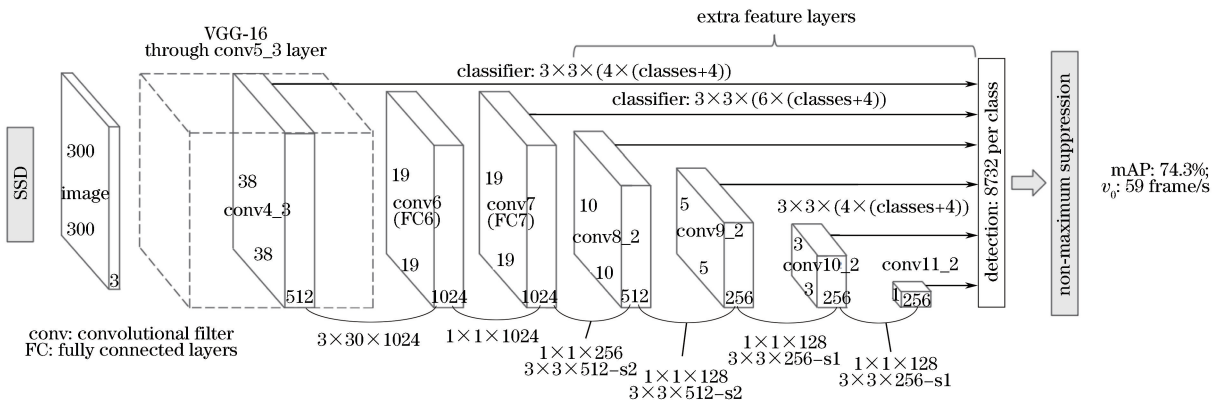


图1 SSD目标检测算法的网络结构

Fig. 1 Network structure of SSD object detection algorithm

2.1.2 默认框机制

SSD使用默认框机制对不同尺度、不同长宽比的目标进行检测。假设用 m 个特征层来检测,第 k 个特征层中默认框对应的尺度 s_k 为

$$s_k = s_{\min} + \frac{s_{\max} - s_{\min}}{m - 1}(k - 1), k \in [1, m], (1)$$

式中: s_{\min} 取值为0.2,代表最底层特征层的尺度为0.2; s_{\max} 取值为0.95,代表最高层特征层的尺度为0.95。使用 a_r 表示不同的长宽比, $a_r = \{1, 2, 3, 1/2, 1/3\}$,则每个默认框的尺寸就可表示为

$$\begin{cases} w_k(a_r) = s_k \sqrt{a_r} \\ h_k(a_r) = s_k / \sqrt{a_r} \end{cases} \quad (2)$$

当长宽比为1时,再增加一个默认框,该默认框的尺度 $s'_k = \sqrt{s_k s_{k+1}}$,则该默认框尺寸为

$$w_k = h_k = \sqrt{s_k s_{k+1}} \quad (3)$$

每个默认框的中心设置为 $(i + 0.5/|f_k|, j + 0.5/|f_k|)$, $|f_k|$ 为第 k 个特征图横向或纵向单元格的数目(特征图中横向单元格数和纵向单元格数相同), i, j 分别代表默认框在横向和纵向的编

号,同时 $i, j \in [0, |f_k|]$,可以看出中心坐标的值均在 0 到 1 之间。此时,默认框与其在原图像上候选框的对应关系为

$$x_{\min} = \left(\frac{i + 0.5}{|f_k|} - \frac{\tau_k}{2} \right) \cdot \tau_{\text{img}}, \quad (4)$$

$$y_{\min} = \left(\frac{j + 0.5}{|f_k|} - \frac{h_k}{2} \right) \cdot h_{\text{img}}, \quad (5)$$

$$x_{\max} = \left(\frac{i + 0.5}{|f_k|} + \frac{\tau_k}{2} \right) \cdot \tau_{\text{img}}, \quad (6)$$

$$y_{\max} = \left(\frac{j + 0.5}{|f_k|} + \frac{h_k}{2} \right) \cdot h_{\text{img}}, \quad (7)$$

式中: τ_k, h_k 为默认框的宽高; $\tau_{\text{img}}, h_{\text{img}}$ 为原始图像的宽高; (x_{\min}, y_{\min}) 代表映射到原始图像上的边界框左上角点; (x_{\max}, y_{\max}) 代表映射到原始图像上的边界框右下角点。

2.1.3 损失函数及参数更新

SSD 的损失函数是置信度损失和位置损失的加权求和,其表达式为

$$L(x, c, l, g) = \frac{1}{N} [L_{\text{conf}}(x, c) + \alpha L_{\text{loc}}(x, l, g)], \quad (8)$$

式中: N 是与真实框相匹配的默认框的个数; α 为用于平衡位置损失和置信度损失的权重系数,设为 1; L_{loc} 为位置回归的损失函数,本文采用 Smooth L1 Loss^[22]; L_{conf} 为置信度损失,本文采用 softmax 函数^[16]; x 用于表征某个预测结果是否与所有真实目标匹配,匹配则 x 为 1,否则 x 为 0; c 为预测结果的置信度; l 为预测边界框的位置信息; g 为真实边界框的位置信息。在训练过程中,该损失函数可以同时提高目标识别的准确度和位置回归的可信度,从而提高目标检测的性能。

SSD 使用了目前常用的优化算法 Mini-Batch Gradient Descent^[23],每次迭代更新取全部训练样本中的一部分进行优化,不需要将全部数据放入内存,从而能有效地提高计算效率。训练时,动量为 0.9,初始学习率为 0.001,批量大小为 8,最大迭代次数为 120000 次。

2.2 梯度聚类模型

梯度聚类的区域建议算法主要由两个主要部分组成:图像梯度提取以及聚类建议区域生成。该算法的输入端与输出端分别是原始图像和送到检测模型的建议区域集,其算法框图如图 2 所示。

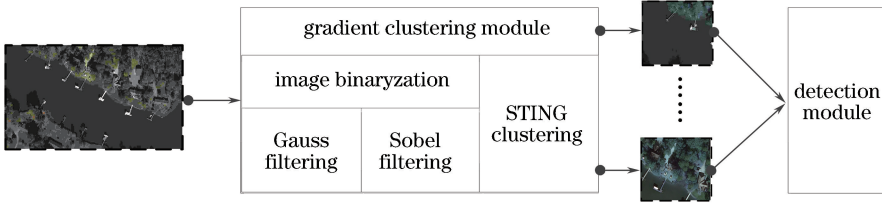


图 2 算法框图

Fig. 2 Structural diagram of algorithm

2.2.1 基于物体边缘特征的图像二值化

在含有大量人造物体的航拍图像中,提取图像梯度可以有效地检测出目标边缘。航拍图像通常以自然背景为主,通过高斯滤除可以有效去除图像中的自然噪声,使图像中人造物体的边缘更加清晰。

设原始航拍图像为 I (RGB 图转化成的灰度图),通过窗口模板尺寸为 $(2k + 1) \times (2k + 1)$ 、标准差为 σ 的高斯滤波器进行滤波,得到滤波后的图像为 I_G 。

目前主要的图像梯度算子有^[24]: Robert 算子、Prewitt 算子、Sobel 算子、LOG (Logarithm) 滤波器以及 Canny 算子。其中,Robert 算子最为简单,且对噪声敏感;Sobel 算子作为 Prewitt 算子的改进版本,具有很好的降噪效果,整体性能优于 Robert 算子和 Prewitt 算子;检测效果最优的 LOG 滤波器和 Canny 算子虽然能够很好地检测出图像边缘的细锐

部分,但是计算更复杂,速度更慢。综合考量提取梯度效果与速度,本文采用 Sobel 算子进行梯度提取。Sobel 算子可表示为

$$S_1 = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, \quad S_2 = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}. \quad (9)$$

(9)式可用于计算图像的一阶梯度。使用 Sobel 算子对高斯滤波后的图像 I_G 求梯度,得到的梯度图为 I_s 。

为了保证 Sobel 梯度运算的质量,本文在 Sobel 滤波前,对高斯滤波后的图像 I_G 周围添加了一圈灰度值为 0 的像素。

对于梯度图像 I_s ,本文采用文献[25]中的方

法:设置全局阈值为 128,对图像进行二值化处理,滤除图像中非人造物体的边缘点,得到图像中人造物体的边缘点的二值图 I_E 为

$$I_E(x, y) = \begin{cases} 0, & I_s(x, y) < 128 \\ 1, & I_s(x, y) \geq 128 \end{cases} \quad (10)$$

2.2.2 基于梯度聚类的区域建议算法

在航拍图像中,人造目标通常会聚集效应,同时人造物体存在大量的边缘,故会出现目标边缘点大量聚集的情况。可以通过对边缘点二值图 I_E 进行聚类,得到多个聚类区域,这些区域就是可能存在目标的感兴趣区域。本文采用已被广泛应用的 STING 算法进行聚类。

STING^[26]算法是王伟等在 1997 年提出的一

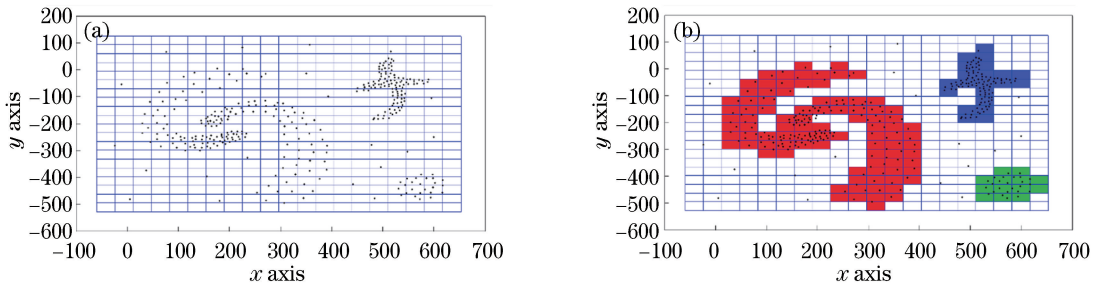


图 3 STING 算法平面空间划分示意图。(a)平面空间栅格化效果图;(b)平面空间聚类效果图

Fig. 3 Schematic of map division in STING algorithm. (a) Rendering of grid raster on map; (b) rendering of clustering on map

对航拍图像的边缘点二值图 I_E 来说,聚类得到的区域就是需要进行目标检测的感兴趣区域。故将得到的聚类区域送入目标检测算法中,以实现航拍图像的目标检测。

3 实验与分析

利用实验验证所提算法的有效性,并将该算法与其他 3 类算法进行比较。实验中,以 SSD 算法作为目标检测算法,在 DOTA 数据集上进行测试,比较梯度聚类 SSD 与其他 3 类算法的实时性与准确率,并分析了数据有效率、离散度和归一化离散度对算法实时性和召回率的影响。

3.1 DOTA 数据集

DOTA 数据集是武汉大学测绘遥感信息工程国家重点实验室夏桂松教授等制作的一个航拍图像数据集,DOTA 包含 2806 张遥感图像,15 个目标类别,图像尺寸约为 4000 pixel×4000 pixel,能很好地反映真实航拍图像的航拍特点,是目前最流行的航拍图像数据集之一。目前,研究人员已经基于该数

种网格聚类算法,可分为两个步骤:单元格划分和查询处理。在单元格划分中算法将对象向量空间划分成一定数量且互不相交的矩形单元,并在单元结构中预先存储数据对象的统计信息。在查询处理中,计算机基于单元格划分中得到的统计信息查询满足要求的单元,并且将这些单元进行连通,得到聚类区域。因为 STING 算法只需要扫描一次基于网络统计信息得到的数据库,所以该算法具有很高的性能。

对二维数据点来说,STING 算法将其构成的平面空间划分成指定大小的网格,如图 3(a)所示,并统计落入每个单元格中点的数量,根据选定阈值指定单元进行连通,得到聚类区域,如图 3(b)所示。

据集组织了一系列的技术竞赛,可以用来比较各类算法的性能。这些竞赛正逐渐成为航拍目标检测领域的 ILSVRC (ImageNet Large Scale Visual Recognition Challenge),推动着相关算法的迭代更新。

3.2 参数选择及检测效果

所提算法需要首先确定的参数有:高斯滤波器核的大小(kernel size)、高斯滤波器的 sigma 值以及网格聚类 STING 算法的网格大小(Grid size)、显著网格的点数阈值(Density),经过具体分析后本文的参数设置如表 1 所示。使用如表 1 所示的参数进行检测时,部分图像的区域建议效果以及检测结果如图 4 所示,其中建议区域效果图中黑色区域为非建议区,其余区域为建议区域。

表 1 所提算法相关参数设置

Algorithm	Kernel size	Sigma	Grid size	Density
Gaussian filter	7×7	1.5	—	—
STING	—	—	300	600

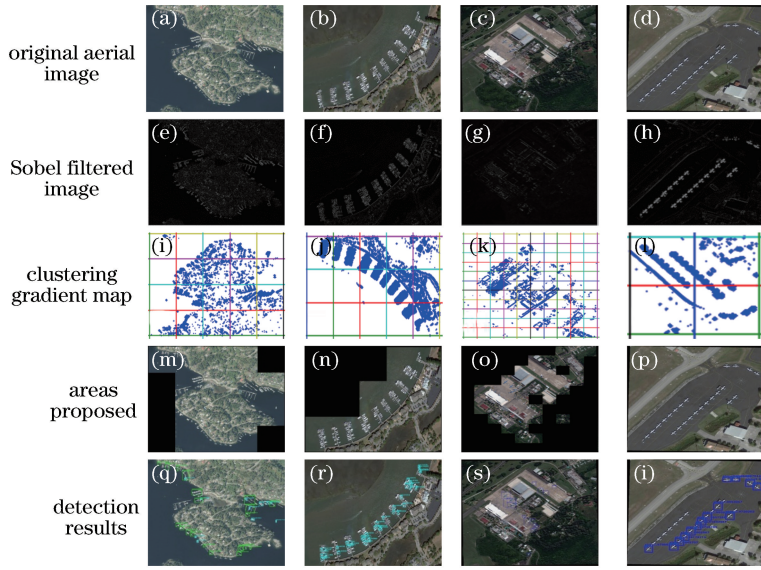


图4 检测结果

Fig. 4 Test results

3.3 实验结果与分析

在DOTA数据集的独立验证集中抽取了80张图片进行测试,其中包含6类目标:飞机(plane)、船(ship)、储油罐(storage tank)、操场(ground track field)、港口(harbor)和桥(bridge)。这些图像的尺寸覆盖范围为2000 pixel×2000 pixel至5000 pixel×

5000 pixel,所使用的实验平台是Ubuntu16.06, CPU为Xeon E5-2673, GPU为GTX1080Ti。表2给出了Faster R-CNN^[15]、YOLO^[27](You Only Look Once)、滑窗法SSD(SW-SSD)与梯度聚类SSD(GC-SSD)的检测速度 \hat{V}_{FPS} 以及平均准确度mAP、准确度AP的对比。

表2 基于DOTA数据集的检测结果

Table 2 Detection results based on DOTA dataset

Method	\hat{V}_{FPS}/s^{-1}	mAP / %	AP / %					
			Plane	Ship	Storage tank	Ground track field	Harbor	Bridge
Faster R-CNN	0.19	51.31	92.55	48.83	68.01	0.5	59.91	38.10
YOLO	0.55	36.59	63.61	75.56	80.35	0	0.05	0
SW-SSD	0.34	48.80	82.46	84.38	76.65	0.5	38.54	10.29
GC-SSD	0.65	46.96	70.14	79.31	74.72	0.5	36.99	20.08

从表2中可以看出,SW-SSD在3种经典算法中,检测效果与检测速度的平衡性较好,其检测速度高于Faster R-CNN,检测精度高于YOLO,但是相比这两种算法的优势还略逊一筹。本文提出的梯度聚类模型旨在提高算法对大尺寸航拍图像中目标的检测速度,保持算法较高的检测精度。由表2可知,相比SW-SSD,GC-SSD在mAP减小3.7%的情况下将目标检测速度提高了91.2%;相比YOLO,GC-SSD的目标检测速度提高了18.1%;相比Faster R-CNN,GC-SSD的检测精度依旧存在一定的差距。总的来说,GC-SSD检测速度显著提高,检测精度小幅下降,但依旧保持在较高的水平,可以说梯度聚类模型对提升大尺寸航拍图像的检测速度是很

有效的。

梯度聚类模型可以提高检测速度的原因是梯度聚类模型倾向于将目标密集的区域作为感兴趣区域,舍弃无目标的冗余区域。这种算法相比其他经典算法,可以在很大程度上减少算法的运算量,因此,GC-SSD算法可以极大地提高大尺寸图像的检测速度。但是GC-SSD算法对某几类目标的检测精度出现明显下降,如飞机、船等,这可能是由于梯度聚类模型在对大尺寸的图像进行梯度运算、聚类操作时,会优先选取目标集中的区域,即梯度点集中的区域,对于那些梯度点达不到显著阈值的部分,梯度聚类模型则会将其舍弃。舍弃的部分有可能存在一些孤立的目标,这种舍弃操作直接导致孤立目标的

漏检,从而影响了目标检测精度。

为此,需在目标检测算法为 SSD 的条件下,分析图像大小、数据有效率等指标对梯度聚类的区域建议算法的实时性和召回率的影响,从理论上分析该算法提高目标检测速度的原因。

3.3.1 实时性分析

为了尽可能保证图像的完整性及检测效果,滑窗法剪裁的检测窗口一般与检测网络的输入大小相同,同时相邻的剪裁窗口存在一定交叠。本文的目标检测算法 SSD 的输入图像为 300 pixel × 300 pixel,因此剪裁窗口大小也为 300 pixel × 300 pixel,两个相邻窗口交叠 50 pixel。以 SSD 的最高帧速度 59 frame/s 为例,应用滑窗法的 SSD 在尺寸从 2000 pixel × 2000 pixel 至 5000 pixel × 5000 pixel 的图像上的帧速如图 5 所示。由图 5 可以看出在大尺寸的航拍图像中,滑窗法 SSD 的帧速很低,最高约能达到 1 frame/s。

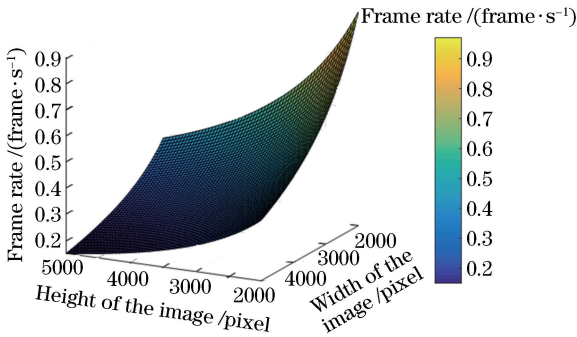


图 5 基于滑窗法的 SSD 的帧速与图像尺寸的关系图

Fig. 5 Relationship between frame rate and image size of SSD based on sliding window method

基于梯度聚类区域建议算法的 SSD 目标检测的帧速度 \hat{V}_{GC-SSD} 主要取决于梯度聚类区域建议算法的帧速度 v_{GC} 、区域建议算法提取的区域数量 N_{APM} 以及 SSD 的帧速度 v_{SSD} 。

$$\hat{V}_{GC-SSD} = \min(v_{GC}, v_{SSD}). \quad (11)$$

所提算法采用 MATLAB 编程,图像梯度是在 GTX1080Ti GPU 上完成的,梯度聚类是在 Xeon E5-2673 处理器上完成的。通过实验,得到了梯度聚类算法耗时 T_C 与聚类数据量 N_C 的关系,如图 6 所示,可以看出梯度聚类算法的用时在 150 ms 以下。

通过拟合,可以得出梯度聚类算法耗时 T_C 与聚类数据量 N_C 的关系为

$$T_C = 7.9421 \times 10^{-5} N_C + 36.5980. \quad (12)$$

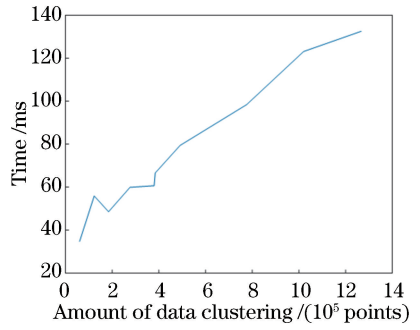


图 6 梯度聚类算法耗时与聚类数据量间的关系

Fig. 6 Relationship between running time and clustering data in gradient clustering algorithm

定义数据有效率 P 为

$$P = \frac{N_C}{N_O}, \quad (13)$$

式中: N_C 为聚类数据量,即原图所包含的人造物体边缘的数量; N_O 为原图中像素总数。数据有效率的含义可以理解为由原图中含有人造物体的像素占原图中总像素的比例,经过实验可以发现,数据有效率通常情况在 8% 以下。

由(12)、(13)式可得,梯度聚类区域建议算法的帧速度 v_{GC-SSD} 的与原图数据量、数据有效率的关系为

$$v_{GC-SSD} = \frac{1000}{T_C} = \frac{1000}{7.9421 \times 10^{-5} N_O P + 36.5980}, \quad (14)$$

如图 7 所示。

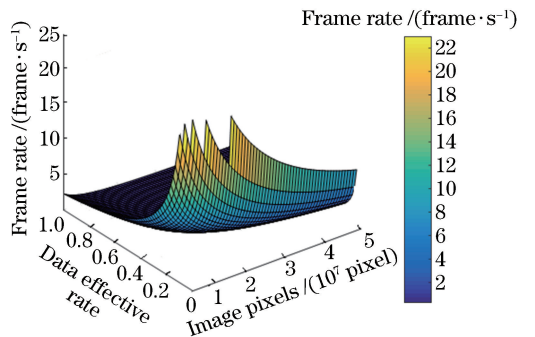


图 7 梯度聚类算法帧速与图像大小、数据有效率间的关系

Fig. 7 Frame rate versus image size and data efficiency in gradient clustering algorithm

由以上分析可知:在不同像素、不同数据有效率的图像上,梯度聚类算法帧速会存在很大差别。由图 7 可以看出,图中有多个峰值,像素越少、数据有效率越小时,梯度聚类算法的处理速度越快。当数据有效率最大达到 8%,图像为最大尺寸 5000 pixel × 5000 pixel 时,梯度聚类算法的帧速最

低(6.8 frame/s),在其他情况下,算法的最高帧速度可以达到100 frame/s以上。

梯度聚类算法分割得到的感兴趣区域块数量 N_{APM} 在 $h \times w$ 的图像上的取值范围为 $N_{APM} \in [0, (h/300) \times (w/300)]$ 。因此,基于梯度聚类区域建议算法中 SSD 的检测帧速度 v_{GC-SSD} 与分割得到的区域块 N_{APM} 的关系为

$$v_{GC-SSD} = v_0 / N_{APM}, N_{APM} \in \left[0, \frac{h}{300} \times \frac{w}{300} \right]. \quad (15)$$

由(15)式可知,当聚类得到的感兴趣区域块越少,检测速度越快。定义图像离散度 Φ 为

$$\Phi = \frac{N_{APM}}{N_s}, \quad (16)$$

式中: N_s 为聚类算法对图像的栅格数量。

由(15)式和(16)式可得

$$v_{GC-SSD} = \frac{v_0}{\Phi N_s}. \quad (17)$$

由(17)式可知:离散度越小,目标检测速度越快;图像栅格化区域块总数越少,目标检测速度越快,其关系如图 8 所示。

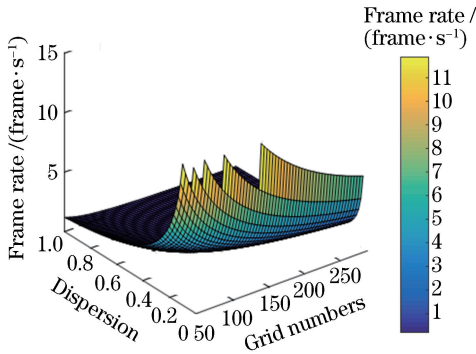


图 8 基于梯度聚类的 SSD 算法检测帧速度与图像离散度、图像栅格数的关系图

Fig. 8 Frame rate versus image pixels and dispersion in SSD algorithm based on gradient clustering

由以上分析可知,基于梯度聚类的 SSD 算法的检测帧速度与目标数量、离散度成反比,因此,在使用基于梯度聚类的 SSD 算法处理目标数量较少且相对集中的航拍图像时,目标检测速度会极大地加快。以 $4000 \text{ pixel} \times 4000 \text{ pixel}$ 的航拍图像为例进行分析,当有效数据量为 5%,离散度为 0.3 时,其帧速度为 1 frame/s,而使用滑窗法 SSD 时,其帧速度为 0.23 frame/s,前者约是后者的 5 倍。

3.3.2 召回率分析

召回率代表在分割出的区域块中包含目标的数

量占整张图像中目标的比率,即

$$R = \frac{O_{APM}}{O_{all}}, \quad (18)$$

式中: O_{APM} 代表梯度聚类算法得到的目标数量, O_{all} 代表一张图像中包含的所有目标数量。

规定当存在将目标被分成多块的情况时,若分割处理的目标块的面积占目标总面积的比例超过 0.5,则代表该目标被选中,否则认为该目标没有选中。考察 large vehicle 和 small vehicle 的召回率情况,根据实验可以得出如图 9 所示的结果。

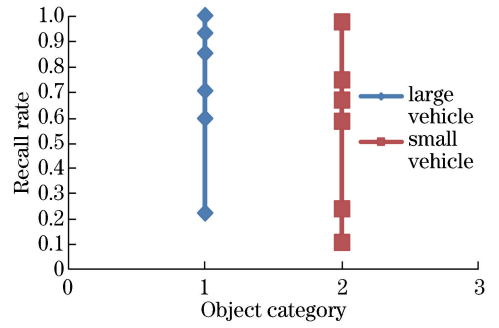


图 9 不同目标的召回率

Fig. 9 Recall rates for different object categories

由图 9 可以看出目标召回率波动很大,从 0.1 到 1.0 都存在,该现象主要由图像中目标的分布位置造成的。定义归一化离散度 D 为

$$D = \sigma_x^2 \sigma_y^2, \quad (19)$$

$$\sigma_x^2 = \frac{\sum (x_a^* - \mu_x^*)^2}{n-1},$$

$$\sigma_y^2 = \frac{\sum (y_a^* - \mu_y^*)^2}{n-1},$$

式中: σ_x^2, σ_y^2 分别代表像素点在横向和纵向的离散度; x^*, y^*, μ^* 分别代表归一化后的 x 值、 y 值及均值,其中 x^*, y^* 的下标 a 代表第 a 个像素点, μ^* 的下标 x, y 分别代表该 μ^* 为归一化后的横向均值和纵向均值; n 代表像素点的总数。

$$\begin{cases} x_i^* = \frac{x_i}{w} \\ y_i^* = \frac{y_i}{h} \\ \mu_x^* = \frac{\sum x_i^*}{n} \\ \mu_y^* = \frac{\sum y_i^*}{n} \end{cases} \quad (20)$$

通过实验可以发现随着归一化离散度逐渐增大,图像的平均召回率逐渐下降,如图 10 所示。

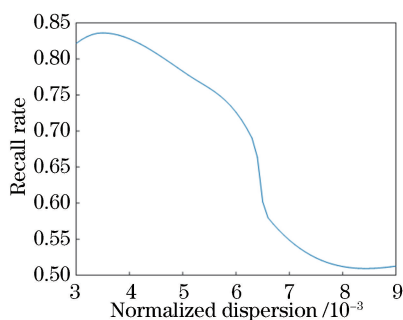


图 10 目标召回率与归一化离散度间的关系

Fig. 10 Relationship between object recall rate and normalized dispersion

由以上分析可知:图像中目标越集中,召回率越高。这是因为当目标更集中时,某个区域的梯度值会更高,该区域更容易被聚类算法选中。滑窗法由于是对全图进行遍历,因此召回率为 100%。

4 结 论

为解决现有目标检测算法对大尺寸航拍图像检测速度慢的问题,提出了基于梯度聚类的区域建议算法。该算法分 2 个步骤:第一步,通过对大尺寸航拍图像中目标边缘进行梯度聚类,得到目标聚集区域,即感兴趣区域;第二步,将这些感兴趣区域送入目标检测算法中进行目标检测,从而实现大尺寸航拍图像的快速检测。该算法与通用的滑窗法相比,能够去除大尺寸图像中无目标的冗余区域,减少检测过程中的计算量,因而能够大幅提高目标检测速度。但是该方法容易漏检孤立目标,因而会造成一定的精度损失。

将梯度聚类模型与 SSD 相结合,对 DOTA 数据集进行测试,结果表明:在大尺寸、目标集中的航拍图像上,本文提出的算法在很大程度上提高了目标检测算法目标检测速度,并且能够保持较高的准确率。本文提出的算法的帧速度是滑窗法的 5 倍以上(图像尺寸为 5000 pixel×5000 pixel)。然而,在目标分散的航拍图像中,该算法的实时性、召回率都会受到影响。在未来的研究工作中,一方面需要进一步改进算法对目标分散的航拍图像的处理效果,以提高算法的稳健性;另一方面考虑使用 GPU 并行技术来实现整体算法的集成,进一步减少算法的运算时间,提高算法的效率。

参 考 文 献

[1] Xiang S M, Fan X W, He N, *et al.* Review on low

light level remote sensing imaging technology [J]. *Laser & Optoelectronics Progress*, 2018, 55(2): 020009.

向世明, 樊学武, 何娜, 等. 微光遥感成像技术研发动态评述[J]. *激光与光电子学进展*, 2018, 55(2): 020009.

[2] Cao Q L, Zhao M, Dong L L, *et al.* Optimization of remote sensing camera and its parameters [J]. *Laser & Optoelectronics Progress*, 2017, 54(3): 032802.

曹旗磊, 赵明, 董丽丽, 等. 航天遥感相机及参数优化方法 [J]. *激光与光电子学进展*, 2017, 54(3): 032802.

[3] Song M Z, Qu H S, Jin G. Weak ship target detection of noisy optical remote sensing image on sea surface [J]. *Acta Optica Sinica*, 2017, 37(10): 1011004.

宋明珠, 曲宏松, 金光. 含噪光学遥感图像海面弱小舰船目标检测 [J]. *光学学报*, 2017, 37(10): 1011004.

[4] Leitloff J, Rosenbaum D, Kurz F, *et al.* An operational system for estimating road traffic information from aerial images [J]. *Remote Sensing*, 2014, 6(11): 11315-11341.

[5] Yao C Q, Chen W. Infrared dim target detection based on improved particle swarm optimization algorithm [J]. *Laser & Optoelectronics Progress*, 2017, 54(11): 111101.

姚成乾, 陈伟. 基于改进粒子算法的红外弱小目标检测研究 [J]. *激光与光电子学进展*, 2017, 54(11): 111101.

[6] Moranduzzo T, Melgani F. Automatic car counting method for unmanned aerial vehicle images [J]. *IEEE Transactions on Geoscience and Remote Sensing*, 2014, 52(3): 1635-1647.

[7] Liu K, Mattyus G. Fast multiclass vehicle detection on aerial images [J]. *IEEE Geoscience and Remote Sensing Letters*, 2015, 12(9): 1938-1942.

[8] Moranduzzo T, Melgani F. Detecting cars in UAV images with a catalog-based approach [J]. *IEEE Transactions on Geoscience and Remote Sensing*, 2014, 52(10): 6356-6367.

[9] Viola P, Jones M. Rapid object detection using a boosted cascade of simple features [C] // *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2001)*, December 8-14, 2001, Kauai, HI, USA. New York: IEEE, 2001, 1: I-I.

[10] Feng X Y, Mei W, Hu D S. Aerial target detection

- based on improved Faster R-CNN[J]. *Acta Optica Sinica*, 2018, 38(6): 0615004.
- 冯小雨, 梅卫, 胡大帅. 基于改进 Faster R-CNN 的空中目标检测[J]. *光学学报*, 2018, 38(6): 0615004.
- [11] Cheng G, Zhou P C, Han J W. Learning rotation-invariant convolutional neural networks for object detection in VHR optical remote sensing images[J]. *IEEE Transactions on Geoscience and Remote Sensing*, 2016, 54(12): 7405-7415.
- [12] Sakla W, Konjevod G, Mundhenk T N. Deep multi-modal vehicle detection in aerial ISR imagery[C] // 2017 IEEE Winter Conference on Applications of Computer Vision (WACV), March 24-31, 2017, Santa Rosa, CA, USA. New York: IEEE, 2017: 916-923.
- [13] Meng Z, Fan X, Chen X, *et al.* Detecting small signs from large images [C] // 2017 IEEE International Conference on Information Reuse and Integration (IRI), August 4-6, 2017, San Diego, CA, USA. New York: IEEE, 2017: 217-224.
- [14] Sommer L W, Schuchert T, Beyerer J. Fast deep vehicle detection in aerial images [C] // 2017 IEEE Winter Conference on Applications of Computer Vision (WACV), March 24-31, 2017, Santa Rosa, CA, USA. New York: IEEE, 2017: 311-319.
- [15] Ren S, He K, Girshick R, *et al.* Faster R-CNN: Towards real-time object detection with region proposal networks[J]. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 2017 (6): 1137-1149.
- [16] Liu W, Anguelov D, Erhan D, *et al.* Ssd: Single shot multibox detector[C] // European Conference on Computer Vision (ECCV), Cham: Springer, 2016: 21-37.
- [17] Everingham M, Eslami S M A, van Gool L, *et al.* The pascal visual object classes challenge: A retrospective[J]. *International Journal of Computer Vision*, 2015, 111(1): 98-136.
- [18] Lin T, Maire M, Belongie S, *et al.* Microsoft COCO: Common objects in context[C] // European Conference on Computer Vision (ECCV), Cham: Springer, 2014: 740-755.
- [19] Deng J, Dong W, Socher R, *et al.* ImageNet: A large-scale hierarchical image database [C] // IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2009, June 20-25, 2009, Miami, FL, USA. New York: IEEE, 2009: 248-255.
- [20] Xia G S, Bai X, Ding J, *et al.* DOTA: A large-scale dataset for object detection in aerial images [C] // IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 18-23, 2018, Salt Lake City, UT, USA. New York: IEEE, 2018: 3974-3983.
- [21] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition[EB/OL]. (2018-08-06)[2015-04-10]. <https://arxiv.org/abs/1409.1556>.
- [22] Girshick R, Donahue J, Darrell T, *et al.* Rich feature hierarchies for accurate object detection and semantic segmentation [C] // IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 23-28, 2014, Columbus, OH, USA. New York: IEEE, 2014: 580-587.
- [23] Ruder S. An overview of gradient descent optimization algorithms [EB/OL]. (2018-08-06)[2017-06-15]. <https://arxiv.org/abs/1609.04747>.
- [24] Gonzales R C, Woods R E. Digital image processing [M]. 2nd ed. Ruan Q Q, Transl. Beijing: Electronic Industry Press, 2002: 474-479.
冈萨雷斯, 伍兹. 数字图像处理[M]. 第二版. 阮秋琦, 译. 北京: 电子工业出版社, 2002: 474-479.
- [25] Han L R. Research on building extraction techniques in complex scene images [D]. Qingdao: Ocean University of China, 2010.
韩丽荣. 复杂场景图像中建筑物提取技术研究[D]. 青岛: 中国海洋大学, 2010.
- [26] Wang W, Yang J, Muntz R. STING: A statistical information grid approach to spatial data mining[C] // International Conference on Very Large Data Bases, 1997: 186-195.
- [27] Redmon J, Farhadi A. YOLO₉₀₀₀: Better, faster, stronger[C] // IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 21-26, 2017, Honolulu, HI, USA. New York: IEEE, 2017: 6517-6525.