基于改进 YOLOv5 的移动应用 GUI 组件识别研究

刘益玮[1] 李 瑛[1] 赵悦彤[1] 石成盛[1]
LIU Yiwei LI Ying ZHAO Yuetong SHI Chengsheng

摘 要 在移动应用开发过程中，自动化测试尤为重要，而 GUI 组件的准确识别是自动化测试执行的基础。针对软件背景构成复杂、GUI 组件目标较小且密集，容易出现误检、漏检等典型问题，提出基于改进 YOLOv5s 的移动应用 GUI 组件识别方法。在 YOLOv5s 主干网络中融入坐标注意力机制，提升小目标组件识别能力；对颈部网络引入 Slim-Neck 结构，在保证检测精度的同时降低模型的复杂度，实现轻量化项目部署。结果表明，改进后 YOLOv5s 算法的 mAP 达到 92.2%，参数量减少 15.7%，相比其他检测模型具有更高的检测精度且更加轻便。

关键词 YOLOv5；GUI 组件；组件识别；坐标注意力机制

doi：10.3969/j.issn.1672-9528.2024.08.003

0 引言

随着移动互联网的发展，移动应用程序在数量和类型上呈现出爆发式增长。为了提升移动应用程序的质量，自动化测试需求也日益增加态势。移动应用程序通常具有图形用户界面（graphics user interface，GUI），用户通过点击屏幕上的组件来完成各种操作，这是用户与移动应用交互的主要媒介。然而，传统的测试方法往往依赖人工编写脚本来定位和操作控件，不仅费时费力，还难以支持跨设备的测试需求 [1]。准确识别和定位移动应用GUI组件的类型和位置是自动化测试执行的基础，这在自动化脚本编写和录制回放中尤为重要 [2]。

录制回放是一种典型的移动应用自动化测试技术。在录制阶段，用户通过界面操作完成特定的测试场景，测试工具会记录用户的点击、输入等交互操作，并转化为可执行的测试脚本或日志。在回放阶段，根据录制的脚本或日志，自动重现之前的用户操作，重复相应的测试场景。虽然录制回放极大简化了测试流程，但是录制的脚本依赖于特定的 GUI 布局，当界面上的组件发生移动、重新排列、尺寸变化时，录制的脚本通常需要相应的调整或更新，以适应这些变化。可以通过 GUI 组件识别技术确定位置和类别解决这个问题。在大多数简单 GUI 组件识别任务中，机器视觉方法，特别是边缘检测算法，能够取得不错的效果。但在面对较为复杂的程序界面和多样化组件时，其识别精度通常较低。深度学习方法能够处理大规模数据集并具有更好的泛化能力，因此越来越多地被用于解决 GUI 组件识别问题。Zhang 等人 [3] 使用 YOLOv3 目标检测算法识别同构 GUI 组件。Cheng 等人 [4] 通过对 YOLO 增加检测层提高对组件小目标识别的成功率，实现组件的识别。张中洋 [5] 通过使用通道注意力机制改善 GUI 组件与背景元素区分困难的问题。郝琳 [6] 将 IoU-guided NMS 应用于 YOLO 中，并增加 IoU（intersection over union）预测分支，提高了预测框定位准确程度。GUI 丰富的背景图案可能会对组件识别带来干扰，不仅是图像组件，其他具有多变设计样式的组件也是如此，所以如何从界面中准确识别组件并分类是一个挑战。由于 GUI 组件在界面中通常密集且目标较小，目前的检测算法对 GUI 组件检测与定位效果并不理想。如果部署到计算资源受限的移动设备上，就需要保证模型轻量且识别准确。现阶段 GUI 组件检测的应用越来越广泛，一个检测精度高、速度快、易扩展的检测方法将更具有竞争力 [7]。

针对以上分析，本文选择 YOLOv5 目标检测算法作为改进前的目标检测算法，选取在检测准确率和模型大小上具有优势的 YOLOv5s 算法。考虑到 GUI 组件目标较小且密集容易出现漏检、错检等问题，引入坐标注意力机制。使用 Slim-Neck 结构，减少模型参数，优化模型性能。

1 YOLOv5 网络

YOLOv5 主要由 Input 输入端、Backbone 骨干网络、Neck 颈部网络、Prediction 输出端四个部分组成，其结构如图 1 所示。输入端对移动应用图像进行预处理，采用 Mosaic 数据增强方法来扩充数据集，同时能自适应图片缩放。骨干

1. 北华航天工业学院 河北廊坊 065000
[基金项目] 校级研究生创新资助项目（YKY-2023-36）；校级研究生创新资助项目（YKY-2024-45）

网络由 Focus 结构、CSP 结构组成，用于实现 GUI 组件的特征提取。颈部网络采用特征金字塔网络（FPN）和路径聚合网络（PAN）相结合的方式，FPN 网络自顶向下与主干特征提取网络特征图融合，促进高层的语义特征与低层特征的融合。PAN 网络自底向上传递目标位置信息，增强多尺度上的定位能力。最后在输出端生成三个不同尺度的特征图，通过计算类别概率和边界框坐标，实现对移动应用 GUI 组件位置和类别的预测。

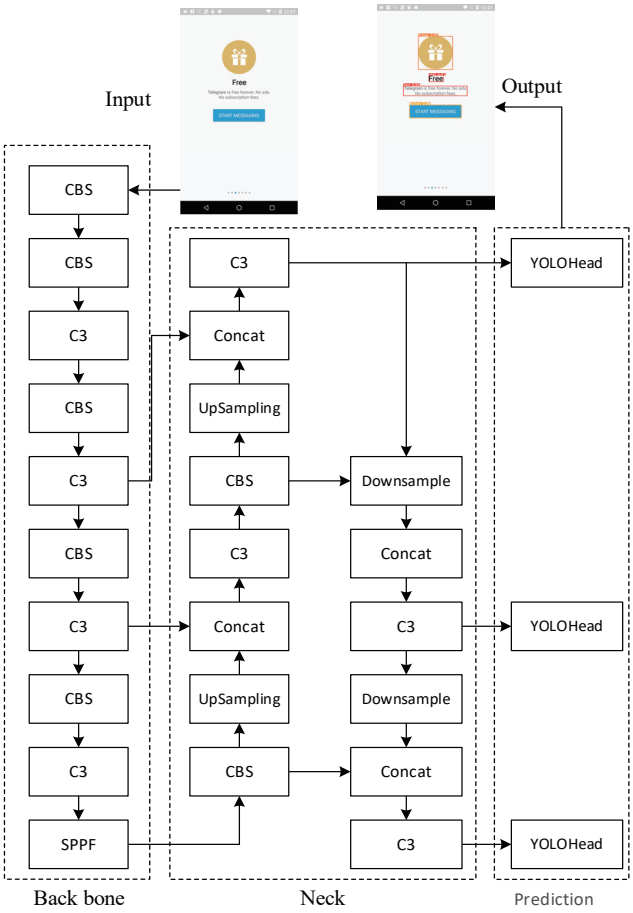

图 1 YOLOv5 结构图

2 基于改进 YOLOv5s 的移动应用 GUI 组件识别

2.1 改进的 YOLOv5s 网络结构

针对目前移动应用 GUI 组件识别中存在的问题，本文在 YOLOv5s 模型基础上对整体网络结构进行了改进，以提高模型对于移动应用 GUI 组件的检测性能。由于移动应用 GUI 组件目标小且密集容易出现漏检的问题，在主干特征提取网络加入 Coordinate Attention 注意力机制，加强对小目标和密集目标的关注度。考虑到模型未来可能在移动设备上部署，在颈部网络引入 Slim-Neck 结构来轻量化网络降低参数量，在保证模型精度和泛化能力的前提下，有效降低模型的复杂程度。改进后的网络结构如图 2 所示。

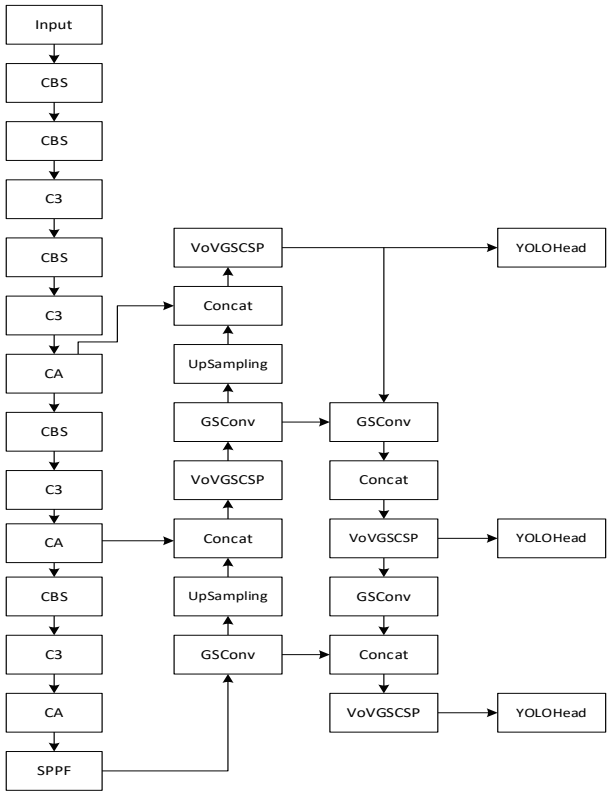

图 2 改进 YOLOv5s 网络结构

2.2 CA 注意力机制

在移动应用 GUI 组件识别中，组件通常相对图像较小且密集，易于受到背景因素影响。为了更精确地检测出界面中的组件，本文引入了 coordinate attention（CA）注意力机制 [8]，使模型更关注 GUI 组件区域的位置信息，以提高对密集小目标组件的检测精度。CA 注意力机制与 SE、CBAM 注意力机制相比，除了关注通道信息外，还捕获了位置信息，在不产生显著计算开销的同时，关注更广泛的区域。CA 注意力机制模块如图 3 所示，其中 C、H、W 分别为特征图的通道数、高度、宽度。

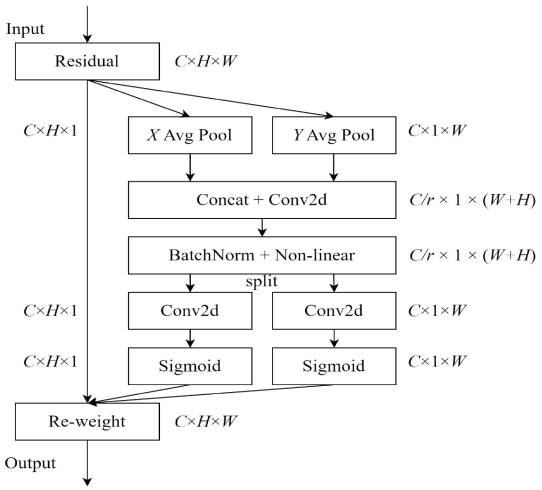

图 3 CA 注意力机制模块

CA 注意力机制为了获取图像宽度和高度上的注意力，将平均池化分解，在水平和垂直方向生成特征图。接着将两个带有方向信息的特征图进行拼接操作，利用 1*1 卷积降维为原来的 *C/r*，其中 *r* 为通道过程中下采样比例。经过非线性激活函数生成中间特征向量包含水平和空间垂直信息。沿着空间维度将特征图切割，用 1*1 卷积结合激活函数 Sigmoid，得到特征图在两个方向上的注意力权重，输出特征图与特征权重相乘获得具有两个方向上注意力权重的特征图。

本文将坐标注意力模块加入 YOLOv5s 模型的 Backbone 中，将 10 层特征提取网络扩展为 13 层的特征提取网络。通过使用坐标注意力机制，让特征提取网络在更大范围上识别到 GUI 组件小而密集的目标，从而提高网络检测性能，改善对 GUI 组件的漏检情况。

2.3 Slim-Neck 结构

考虑到识别 GUI 组件完成移动应用测试需要将目标检测模型部署到资源受限的移动设备上，且在检测速度和准确率上有一定的要求，所以采用 Slim-Neck 结构 [9]。这个结构既减轻了模型复杂度，又保持了精度，适合实现轻量化项目部署。

Slim-Neck 网络结构如图 4 所示，为满足实时检测需求，采用 GSConv 构建 VoVGSCSP 模块。GSConv 采用标准卷积（SC）、深度可分离卷积（DSC）以及 Shuffle 相结合的方式，其计算成本仅是标准卷积的 60% ～ 70%，既能轻量化模型，也能保持准确率。GSConv 首先输入进行一个普通卷积的下采样，然后使用 DWConv 深度卷积，并将 SC 和 DSC 的结果拼接起来，最后进行 shuffle 操作。GSConv 旨在以较低的时间复杂度尽可能地保留连接，以提高神经网络的性能和效率。其适合在颈部网络处理特征图融合阶段中使用，因为在这个阶段，冗余和重复信息相对较少，不需要对特征图进行进一步压缩。VoVGSCSP 结构可以通过一次聚合来融合不同级别的特征图，从而实现更加精细和全面的特征表示。

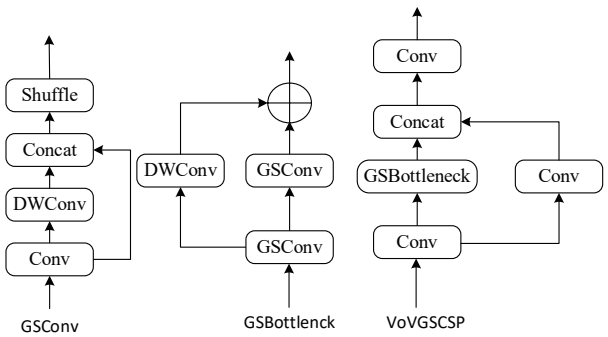

图 4 Slim-Neck 结构

本文将 YOLOv5 颈部网络替换为 Slim-Neck 结构，以实现网络轻量化，使其适合部署于资源受限的设备。通过此方法优化性能并保证实时性，提升了模型的实用性。

3 实验结果与分析

3.1 数据集

本文实验从 RICO 数据集 [10] 中筛选不重复的高质量图片 2290 张，使用 LabelImg 软件标注了常见的 7 类，分别为文本 text、图标 Icon、图片 Image、按钮 Button、输入框 Input、复选框 Checkbox、单选框 Redio Button。

3.2 实验环境

本实验在 Python3.8，CUDA12.3，PyTorch 版本为 2.1.0。所有模型均在 NVIDIA 4060tiGPU 上进行训练和测试，数据集按照 8:2 划分为训练集和验证集，batchsize 设置为 16，使用 SGD 优化器，学习率设置为 0.01。总共进行 200 次迭代。

3.2 消融实验

为进一步验证本文所采用改进策略的有效性，将各个改进策略分别加入原模型，进行消融实验。实验结果如表 1 所示，从表 1 中可以看出原始的 YOLOv5s 模型的 mAP 为 91.1%，参数量为 7.0 MB。在引入 SlimNeck 后，没有提升太多检测性能，但是参数量减少到 5.8 MB，mAP 为 91.3%。在进一步引入 CA 注意力机制后，mAP 达到 92.2%，参数量为 5.9 MB。改进的策略组合在一起可以提升模型检测性能并保证模型的轻量化。

表 1 消融实验

Model	加入 Slim-Neck	加入 CA	mAP/%	GFLOPs	参数量/MB
YOLOv5s	×	×	91.1	16.0	7.0
优化模型 1	√	×	91.3	12.8	5.8
优化模型 2	√	√	92.2	12.9	5.9

3.3 对比实验

对比实验是为了更好展现改进模型的优势，该实验是将本文改进后的模型与 FasterRCNN、SSD、YOLOv3、YOLOv4、YOLOv5s 进行对比，使用相同数据集进行训练验证，如表 2 所示。相较于两阶段的 Faster-RCNN 和一阶段的 SSD、YOLOv3、YOLOv4，目标检测算法中 YOLOv5s 模型更为轻量，本文方法在此基础上权重文件减少了 2.1 MB。改进的方法对比 Faster-RCNN、SSD、YOLOv3、YOLOv4、YOLOv5 的 mAP 分别提高 12.1、42、10、5.7 和 1.1 个百分点。综上，本文提出的改进算法对 GUI 组件识别具有较强的能力，并且计算量更少。

表 2 对比实验

Model	Precision /%	Recall /%	权重 /MB	FPS /(帧·s^{-1})	mAP /%
Faster-RCNN	52.76	90.76	108	15	80.1
SSD	65.22	42.58	93.6	35	50.18
YOLOv3	87.4	69.78	235	51	82.2
YOLOv4	89.2	79.64	244	38	86.5
YOLOv5s	89.5	89.4	13.7	138	91.1
论文方法	89.4	89.4	**11.6**	118	**92.2**

3.4 检测结果分析

为了更好验证检测效果，本文使用原始 YOLOv5s 模型与改进的 YOLOv5s 模型从数据集中选取部分图片以及选取目前应用截图进行可视化对比验证，如图 5 所示。上面为使用原始 YOLOv5s 网络检测的结果，下面为改进后 YOLOv5s 网络的检测结果。根据可视化结果可以看出，第一组图改进前对文本识别丢失误认为背景，改进后可以识别。第二组图片改进前识别“第 1 阶段：”文本拆分为两段，改进后的可以识别。最后一组是复杂场景，改进前关注和发现的文本未识别，改进后可以识别。本文改进后的方法对界面组件小目标检测效果良好，但是在面对复杂场景时还存在错误和漏检问题。

图 5 测试结果对比图

4 结论

本文针对移动应用 GUI 组件目标小且密集，容易导致漏检错检问题，提出基于改进 YOLOv5s 的 GUI 组件识别算法。首先，在 YOLOv5s 的主干网络引入坐标注意力机制，提高网络在更大区域对较小组件的信息提取能力。然后，在颈部网络采用 Slim-Neck 网络结构，在保证模型检测精度和泛化能力的同时降低模型计算量与参数量。相比于原 YOLOv5s 模型，改进后的模型平均精度达到 92.2%，参数量减少了 15.7%。将本文改进后的算法与其他算法对比后，实验结果表明改进后的 YOLOv5s 模型综合表现最佳，能够有效识别移动应用 GUI 组件的类别和位置，为移动应用自动化测试提供助力。

参考文献：

[1] 李聪，蒋炎岩，许畅．基于 GUI 事件的安卓应用录制重放关键技术综述 [J]. 软件学报 ,2022,33(5):1612-1634.

[2] 张文烨．基于图像识别的移动端应用控件检测方法 [J]. 计算机应用 ,2020,40(S1):157-160.

[3]ZHANG T, LIU Y, GAO J, et al.Deep learning-based mobile application isomorphic GUI identification for automated robotic testing[J].IEEE software,2020,37(4):67-74.

[4]CHENG J, TAN D, ZHANG T, et al.YOLOv5-MGC:GUI element identification for mobile applications based on improved YOLOv5[J].Mobile information systems, 2022, 2022: 8900734.1-8900734.9.

[5] 张中洋．基于深度学习的移动应用图形用户界面组件识别方法研究 [D]. 成都：四川大学 ,2021.

[6] 郝琳．面向 GUI 控件识别的深度学习图像匹配算法研究 [D]. 廊坊：北华航天工业学院 ,2023.

[7] 邵耀堃．基于深度学习的图形用户界面组件检测方法研究 [D]. 广州：华南理工大学 ,2022.

[8]HOU Q, ZHOU D, FENG J.Coordinate attention for efficient mobile network design[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. Piscataway: IEEE, 2021:13713-13722.

[9]LI H, LI J, WEI H, et al.Slim-neck by GSConv: a better design paradigm of detector architectures for autonomous vehicles[J]. (2022-06-06)[2024-01-02].https://arxiv.org/abs/2206.02424.

[10]DEKA B, HUANG Z, FRANZEN C, et al. Rico: a mobile app dataset for building data-driven design applications[C]// Proceedings of the 30th annual ACM symposium on user interface software and technology.New York: ACM, 2017: 845-854.

【作者简介】

刘益玮（2000—），男，河北承德人，硕士，研究方向：计算机技术。

李瑛（1982—），通信作者（email：lan_yanjing@163.com），女，陕西旬阳人，硕士，副教授，研究方向：智能软件测试、软件质量保证。

（收稿日期：2024-05-15）