**以下内容为YOLO-V3技术的背景介绍。**

之前我国主要采用人工方法检查零件中的缺陷，这种方法不仅经常发生漏检现象，而且极度依赖工人的经验，需要耗费大量的人力和物力，已经不适应如今产业的发展。随着深度学习和神经网络的兴起，国内外学者提出了许多基于神经网络的目标检测和目标跟踪算法，并且已有相关研究将其用于处理零件图像， 如通过卷积神经网络识别区域等零件状态等，获得了良好的检测效果。目前应用比较广泛且基于深度卷积神经网络的目标检测算法可以分为两类: 一类是基于候选区域的目标检测算法，如Ｒ－CNN( Ｒegion－Conventional Neu- ral Network)、Fast Ｒ－CNN等，这类算法首先获取图像中可能存在目标的子区域，接着将所有的子区域作为输入，并通过深度卷积神经网络提取目标特征，最后进行类别检测和边框修正; 另一类是基于端对端学习的目标检测算法，如SSD( Single Shot Multibox Detector )、YOLO ( You Only Look Once)、YOLO V3等，此类算法不需要提取候选区域，只要给定输入图像，就可以直接利用该算法检测出图像中的目标类别和目标边框，大大加快了检测速度。本文将采集到的现场环境下的图像样本制作成零件缺陷检测数据集，以 YOLO V3 模型为基础，将零件的各类缺陷作为训练和检测的目标。

1. **YOLOv3介绍**

YOLOv3（You Only Look Once，第3版）是一种实时对象检测算法，可识别视频、实时提要或图像中的特定对象。YOLO 使用深度卷积神经网络学习的特征来检测物体。YOLO 的 1-3 版本由 Joseph Redmon 和 Ali Farhadi 创建。

YOLO 的第一个版本创建于 2016 年，而本文中广泛讨论的第3版是在两年后的 2018 年制作的。YOLOv3 是 YOLO 和 YOLOv2 的改进版本。YOLO 是使用 Keras 或 OpenCV深度学习库实现的。

AI程序使用对象分类系统将类中的特定对象视为感兴趣的主题。系统将图像中的对象分类成组，其中具有相似特征的对象被放置在一起，而其他对象则被忽略。作为目标检测器的典型特征，卷积层学习的特征被传递给分类器，该分类器进行检测预测。在 YOLO 中，预测基于使用 1×1 卷积的卷积层。YOLO 之所以被命名为“you only look once”，是因为它的预测使用了 1×1 的卷积；预测图的大小正好是它之前的特征图的大小。

YOLO 是一种卷积神经网络 (CNN)，用于实时执行对象检测。CNN 是基于分类器的系统，可以将输入图像处理为结构化的数据数组并识别它们之间的模式（如下图1所示）。YOLO 具有比其他网络快得多的优势，并且仍然保持准确性。

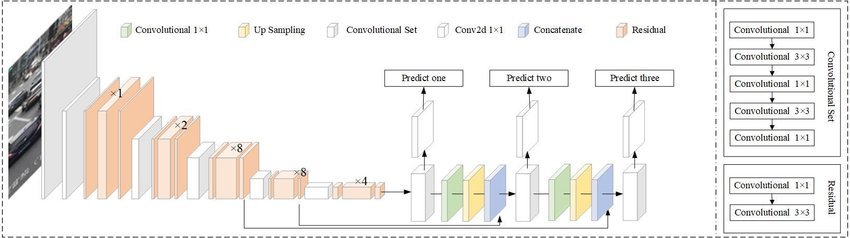


Figure 1: YOLO v3 的工作原理

它允许模型在测试时查看整个图像，因此它的预测是由图像中的全局上下文通知的。YOLO 和其他卷积神经网络算法根据区域与预定义类的相似性“评分”区域。

高分区域被记录为对与它们最密切相关的任何类别的阳性检测。例如，在实时交通流中，YOLO 可用于检测不同类型的车辆，具体取决于视频的哪些区域与预定义的车辆类别相比得分高。

1. **YOLOv3架构**

YOLOv3算法首先将图像分成网格。每个网格单元预测在上述预定义类中得分高的对象周围的一些边界框（有时称为锚框）。

每个边界框都有一个相应的置信度分数，表示它假设预测应该有多准确，并且每个边界框只检测一个对象。边界框是通过从原始数据集中对地面实况框的维度进行聚类来生成的，以找到最常见的形状和大小。

其他可以实现相同目标的类似算法是 R-CNN（2015 年制造的基于区域的卷积神经网络）和 Fast R-CNN（2017 年开发的 R-CNN 改进）和Mask R-CNN。

然而，与 R-CNN 和 Fast R-CNN 等系统不同，YOLO 被训练为同时进行分类和边界框回归。

1. **YOLOv3的新功能**

YOLOv3 和旧版本在速度、精度和类的特异性方面存在重大差异。YOLOv2 和 YOLOv3 在准确性、速度和架构方面是天壤之别。YOLOv2 于 2016 年问世，比 YOLO v3 早了两年。

以下部分将概述 YOLOv3 中的新功能。

* 1. **更快的速度**

YOLOv2 使用 Darknet-19 作为其骨干特征提取器，而 YOLOv3 现在使用 Darknet-53。Darknet-53 也是由 YOLO 的创造者 Joseph Redmon 和 Ali Farhadi 制作的骨干网。

Darknet-53 有 53 个卷积层，而不是之前的 19 个，这使得它比 Darknet-19 更强大，并且比竞争的骨干网（ResNet-101 或 ResNet-152）更高效。（如下图2所示）

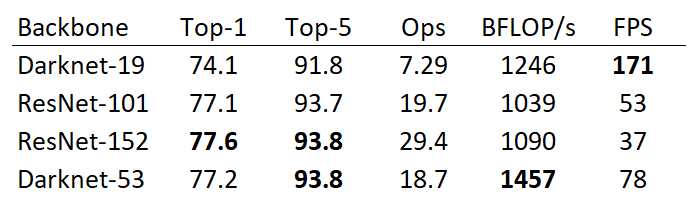


Figure 2: 骨干比较。各种网络的精度、数十亿次操作 (Ops)、每秒十亿次浮点操作 (BFLOP/s) 和每秒帧数 (FPS)

使用Redmon 和 Farhadi 在 YOLOv3 论文中提供的图表，我们可以看到 Darknet-52 比 ResNet101 快 1.5 倍。所描绘的准确性也不需要在暗网主干之间的准确性和速度之间进行任何权衡，因为它仍然与 ResNet-152 一样准确，但速度快了两倍。

YOLOv3 在平均精度 (mAP) 和联合交集 (IOU) 值方面也是快速且准确的。它的运行速度明显快于其他性能相当的检测方法。（如下图3所示）

此外，您只需更改模型的大小即可轻松地在速度和准确性之间进行权衡，而无需重新训练模型。

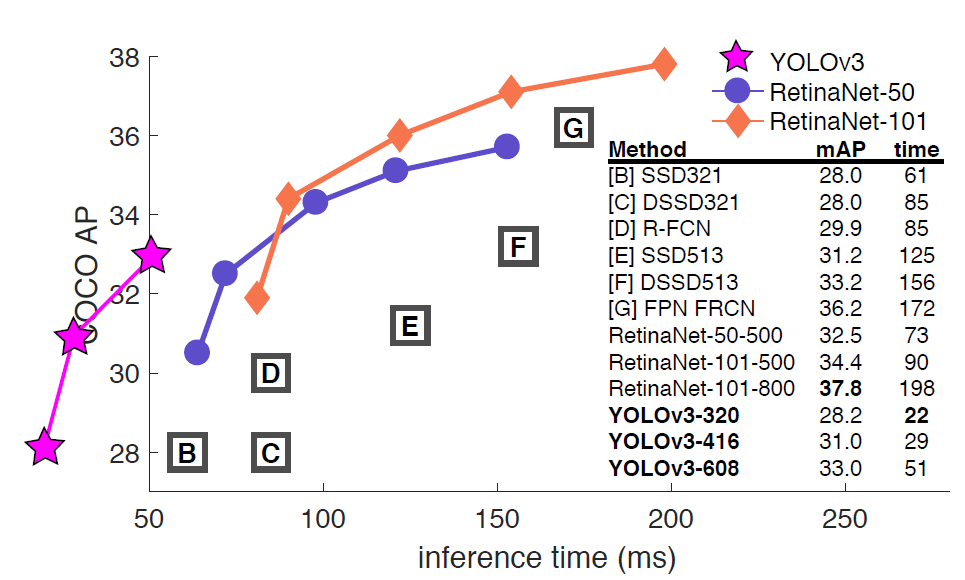


Figure 3: YOLOv3 比使用 M40/Titan X GPU 的性能相当的其他检测方法运行得更快

* 1. **更高的精度**

下图（如下图4所示）显示了使用各种算法和主干检测小、中、大图像的平均精度 (AP)。AP 越高，该变量的准确性就越高。

YOLOv2 对小物体的精确度是其他算法无法比拟的，因为 YOLO 在检测小物体时非常不准确。它的 AP 为 5.0，与 RetinaNet (21.8) 或 SSD513 (10.2) 等其他算法相比相形见绌，后者的小物体 AP 是第二低的。

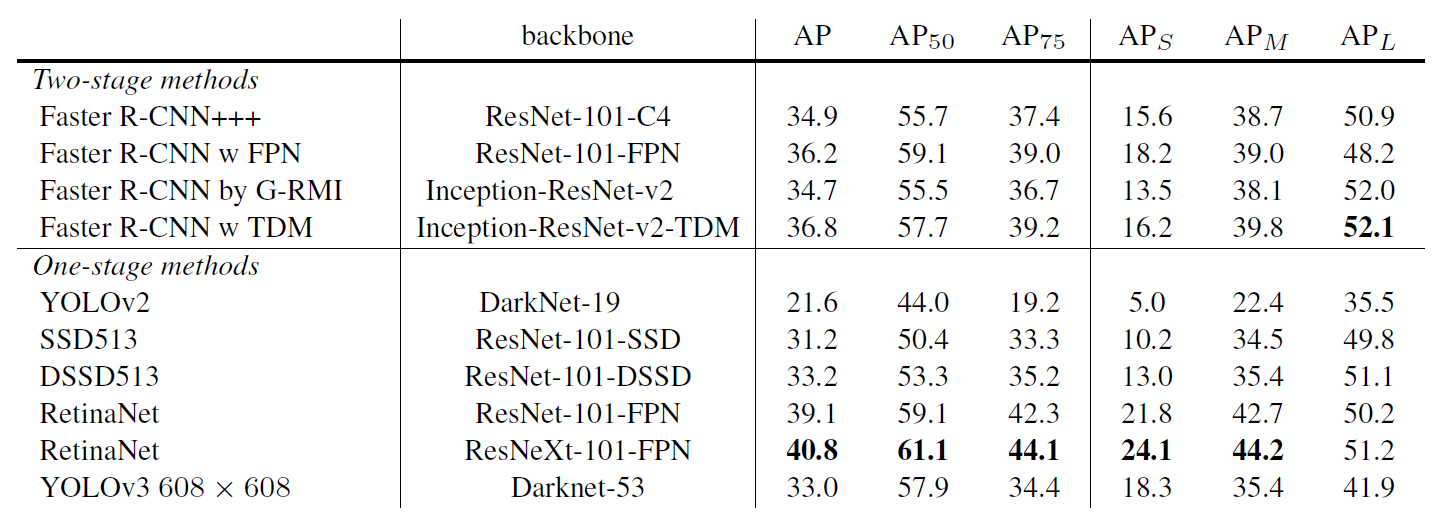


Figure 4: YOLOv3 对不同对象大小的比较显示了 AP-S（小对象大小）、AP-M（中等对象大小）、AP-L（大对象大小）的平均精度 (AP)

YOLOv3 将小物体的 AP 提高了 13.3，这是 YOLOv2 的巨大进步。但是，所有对象（小、中、大）的平均精度（AP）仍然低于 RetinaNet。

* 1. **更加特殊的类**

新的 YOLOv3 在训练期间使用独立的逻辑分类器和二元交叉熵损失进行类预测。这些编辑使得使用复杂的数据集（例如 Microsoft 的开放图像数据集 (OID)）进行 YOLOv3 模型训练成为可能。OID 包含数十个重叠标签，例如数据集中图像的“人”和“人”。

YOLO v3 使用多标签方法，该方法允许类更具体，并且对于单个边界框来说是多个。同时，YOLOv2 使用了 softmax，这是一种将数字向量转换为概率向量的数学函数，其中每个值的概率与向量中每个值的相对比例成正比。

使用 softmax 使得每个边界框只能属于一个类，有时情况并非如此，尤其是对于像 OID 这样的数据集。