**Yolo V5 Deep convolutional neural network for Advanced traffic sign recognition with synthetic training data developed on Various GAN**

**Deep convolutional neural network for enhancing traffic sign recognition developed on Yolo V5**

**Abstract：**在当今天时代，含有多个隐藏层的深度学习神经网络在多个领域都得到了广泛的应用， 相比于传统机器学习方式，这种方式具有更加强大的特征学习和特征。而随着物流行业的发展和自动驾驶的盛行，交通标志识别吸引了越来越多的研究者的目光。首先我们通过MSRCR算法实现图像的增强，并且针对上述挑战，在这片论文中，我们从训练方式和网络结构的角度，基于YOLOv5实现了对交通标志的自动分类。此外，我们并且GTSRB，GTSDB，CCTSDB数据集上的校验，将这些数据集作为目前检测的训练集和测试集。

**Keywords:** deep learning, YOLO algorithm, traffic signs, object recognition

**Chapter 1 Introduction**

随着信息技术的发展，深度学习技术，特别是基于视觉的目标检测技术越来越多的渗透到了人们生活的方方面面。出入公众场所的人脸识别校验，智慧城市的汽车检测，街道摄像头的监控，都与基于视觉的目标检测息息相关。

早在上世纪60年代，有关学者就已经开始了对基于视觉的目标检查的研究[1]。早期学者通过级联分类器，支持向量机等方式实现对目标的鲁棒实施检测，这种基于Scale-invariant feature transform, SIFT的模式在早期被视为一种有效的方式，但是受限于技术限制（基于SIFT提取的特征对模型的好坏有着相当重要的影响），检测准确率往往不高，模型的泛化能力比较差，难以应用在工业乃至商业领域。

在2006年，Geoffrey Hinton 和他的学生Ruslan Salakhutdinov在Science上发表题目为“Reducing the dimensionality of data with neural networks”的文章[2]，这标志着深度学习的正式开始。含有多个隐层的深度学习神经网络具有非常强大的特征学习功能，可以通过训练模型提取特征从原始输入数据具有更抽象和更本质的表述。这种通过深度学习的训练神经网络的方式首先被应用于语音识别领域，相较于传统方式，准确率，precision和recall都大幅提升，甚至达到了20%-30%之巨。仅仅不到一年之后，卷积神经网络就在大规模神经网络上得到了研究者的关注，互联网公司巨头例如谷歌，微软也相继投入大量资源，布局深度学习。

交通，作为一个国家的支柱型重要基础产业。而目前，随着自动驾驶技术的飞速发展和生活水平的提高，汽车已经成为人们日常出行的重要交通工具，智能交通也得到了越来越多的关注。而交通标志在智能交通网络中起着至关重要的作用，这些标志用文字和符号向驾驶员展示了当前路段的交通状况[3]。通过深度学习技术识别交通灯是一个very promising的领域，在智能交通领域中能够起到避免道路碰撞、调度车流的作用，同时在自动驾驶方面也起着至关重要的作用。

但是由于交通标志的多样性，以及道路，天气状况的多样性，同时，包含亮度，颜色，遮挡等问题，交通标志检测这一领域的基础研究具有比较大的挑战性。交通信号灯通常是较小的图像，通常仅仅占有图片非常小的部分，并且天气状况非常复杂，有时候会是多云，下雨，晴天等多样的状况，甚至可能会有广告牌的遮挡，这给研究带来了比较大的难度。

目前，大多数相关算法都只为检测少量类别而开发，并且难以客服自然，闪电，风，雨等自然环境因素影响，此外，摄像头捕捉的画面模糊也没有考虑在内，这与实际情况严重不符。而且部分算法只关注分类问题而忽略了交通标志位置的预测问题，难以应用到工业乃至商业上。

本文围绕深度学习的YOLO（You Only Look Once）系列算法在交通检测中的应用开展了一系列实证分析，建立了基于CNN的交通标志识别模型，并对提升实时检测的效率和精度做出相应的总结。我们的目的是开发一种可以有效识别交通标志的深度学习神经网络，为了达成这一目标，我们在GTSRB（German Trafﬁc Sign

Recognition Benchmark）, GTSDB（German Trafﬁc Sign Detection

Benchmark）, CCTSDB(Changsha University of Science and Technology Chinese traffic sign detection benchmark)上进行了训练，并且取得了不错的成绩。

**Chapter 2 Related Work**

**2.1 卷积神经网络**

在过去的10年中，CNN网络实现了ground-breaking breakthroughs，并且在多个领域都取得了突破性的进展。CNN is a feedforward neural network, 为了处理二维输入数据，从而特殊设计了该多层人工神经网络，其中网络中的每层都由多个独立的神经元组成。

卷积神经网络将原始图像的像素映射到一个可以区分维度的空间数据中，打破底层像素和高层语义之间的语义鸿沟至关重要的一步。作为一种处理输入数据而特殊设计的多层神经网络，每一层都含有若干个神经元，每两层神经元之间都互相设有链接，而同一层之间的神经网络则彼此之间没有链接。同时，模型的容量可以通过网络的深度和广度来调整，同时，通过卷积层提取到的特征输入至分类器，并且实现最后的预测结果[4]。

A convolutional neural network comprises an input layer, hidden layers, and an output layer. Fig. 1 shows an overview of the architecture. Hidden layers often include convolutional layers and pooling layers. The input layer typically represents the picture’s pixel matrix in neural network image processing. 图像输入后在convolutional layer通过与3个卷积核和可加偏置向量进行卷积运算，产生3个特征映射图。

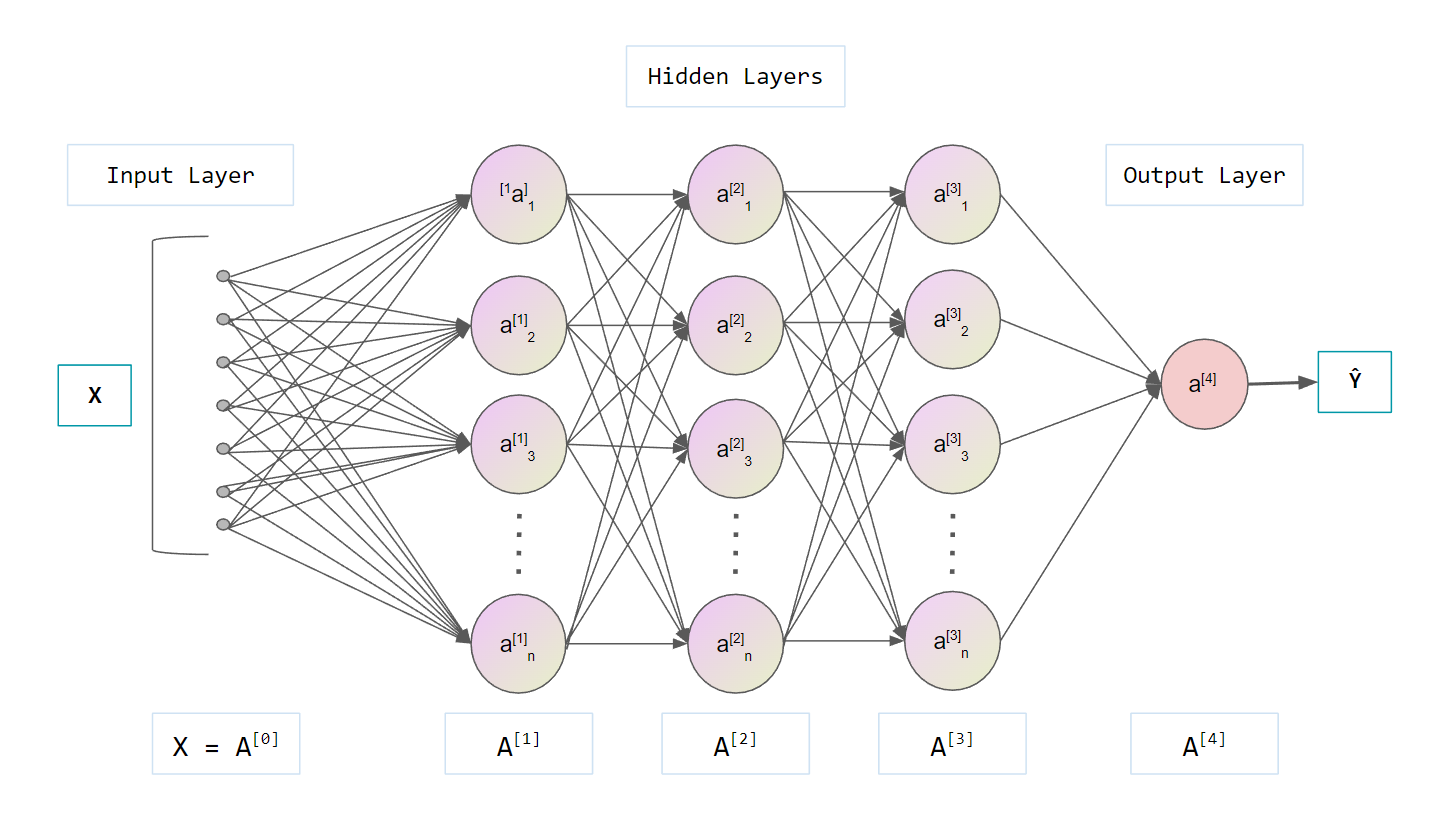


Fig 1: An example of CNN network (Draw with ml-visual)

**2.2 目标检测算法**

基于深度学习的目标检测领域具有包含传统的Two-Stage和One-Stage算法。前者以R-CNN，Faster-RCNN 为代表，后者以YOLO为代表。R-CNN[5]是第一个提出的能够达到工业级精度的Two-Stage算法，但是存在检测缓慢，不能满足快速相应的要求。而随着One-Stage算法的提出，目标检测的速度大幅度提高，例如YOLOv3.

**2.3 YOLOv5**

**Chapter 3 Architecture of Proposed Network**

In this network, we mainly trained three models. Model-1 was trained on GTSTRB （German Traffic Sign Recognition Benchmark） which has more than 50 thousands RGB images in total with 32909 for the train set and 12631 for the test set . The image in this dataset can be classified as 43 classes and it contains the same image in the multiple conditions. The 43 classes of traffic sign include all the traffic signs required by the german law (SL20Kmhr, SL30Kmhr, SL50Kmhr, SL60Kmhr, SL70Kmhr, SL80Kmhr, EOSL80Kmhr, SL100Kmhr, SL120Kmhr, NoPassing, NPFforV35Tons, RIWATNIntersection, PriorityRoad, Yield, Stop, NoVehicles, V35TonProhibited, NoEntry, GenCaution, DangCurveLeft, DangCurveRight, DoubleCurve, BumpyRoad, SlipperyRoad, RoadNarrowRight, RoadWork, TrafficSignals, Pedestrians, ChilCrossing, BzCrossing, BewareIce, WildAnmCrossing, EndOfAllSpeed, TurnRightAhead, TurnLeftAhead, AheadOnly, GoStrOrRight, GoStrOrLeft, KeepRight, KeepLeft, RoundAbtMandatory, EndOfNoPassing, EndOfNoPassingV35).

The annotation for this dataset is given in a single text file and 我们使用Python的Numpy和Pandas方法将其转化为YOLO格式，the trained image in YOLO set comes in this format: class ID, center of bounding box in x direction, center of bounding box in Y direction, bounding box width, bounding box height. In the label file, each row represent 一个目标，所以我们可能会有多个行。We calculate using the following formula:

Xcenter = (Xmax + Xmin)/W/2

Ycenter = (Ymax + Ymin)/H/2

where, W and H represent the width and height of the bounding box in the real RGB image from the GTSRB dataset respectively.

After the transformation work, we got the annotation and image respectively. And they are stored in two different folders (image, label) which have train and test subfolders. Therefore, every image in this preparation dataset would have a single text file corresponding to it which contains the information of the bounding box from it.

我们的YOLO 网络由输入端，Backbone，Neck和Head组成。在输入端，为了实现更加复杂的图片背景，使用Mosaic数据增强将四张图片进行组合，这样，我们训练出来的网络能够应对更加复杂的自然环境以及交通标志所在的环境。

Backbone部分主要包含了包含了 BottleneckCSP和 Focus 模块，前者能够大幅降低网络的计算量，同时维持网络的精度几乎不变甚至有所降低，而Focus模块对图片进行切片操作，通过卷积得到下采样卷积层which can also 减少计算量加快网络速度. YOLO模型的卷积操作与常规意义上的卷积有所不同，而是采用CBL进行作用产生卷积。上述的操作能够让我们从YOLO中提取特征层，下面是其中一个特征层：

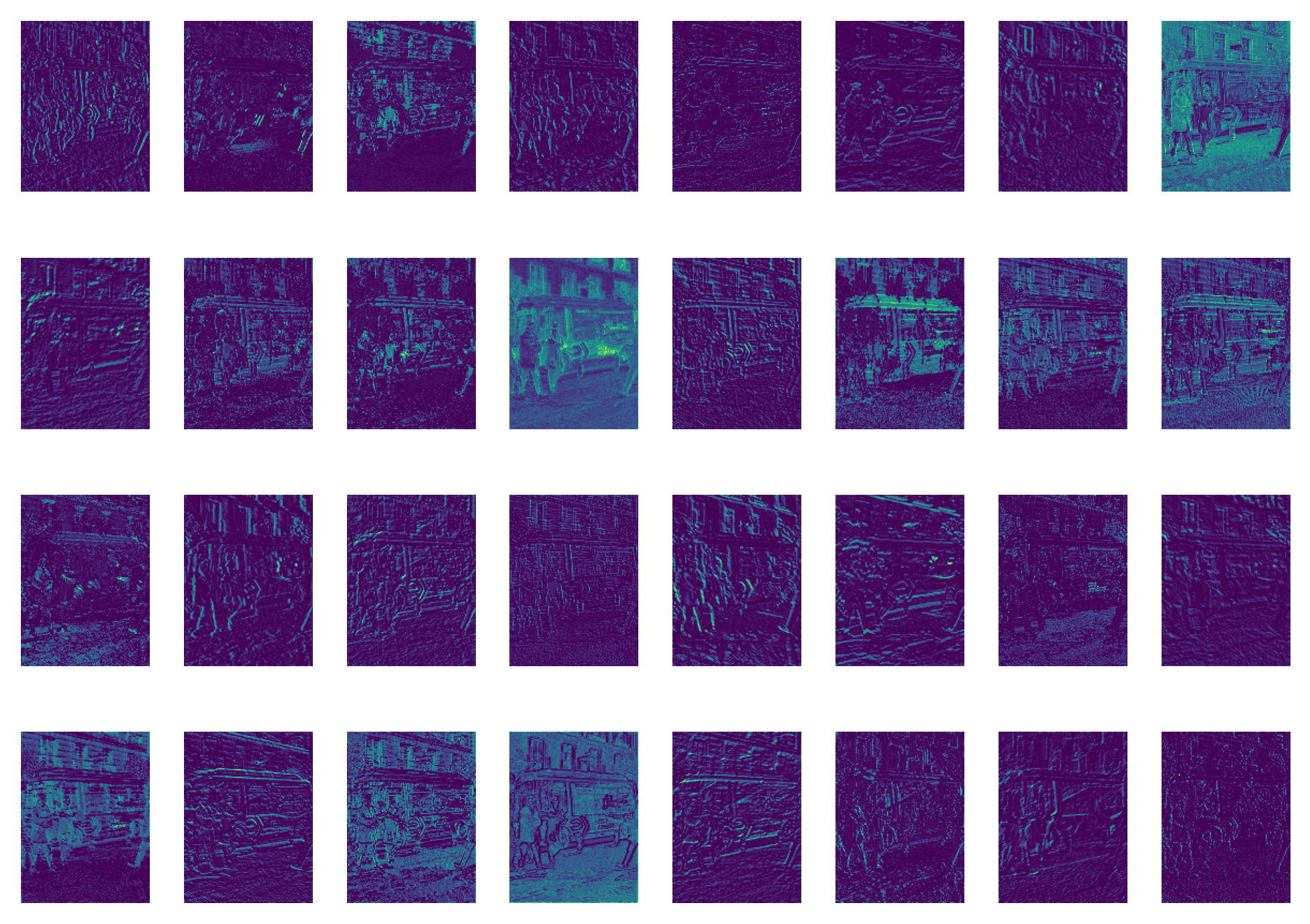


Fig: Feature Map 3/24

Neck部分的组件相比较于Backbone十分单一，由CBS，UpperSample和Concat组成，同时，沿用了FPN+PAN的结构。这个Componenets对特征进行一波混合与组合，并且把这些特征传递给预测层。

Head中则会将目标对象的类别概率、对象得分和该对象边界框的位置以一个向量的形式输出出来，检测层中输出的特征向量最后会被还原到原始上。

The activation functions we use include leakyReLU and Sigmoid, the middle hidden layer uses the leakyReLU activation function, and the final detection layer uses the Sigmoid activation function.

The CNN architecture of our model is adapting from the YOLOv5 paper.

Applying the YOLOv5 with many tricks on this dataset, the confusion matrix of this dataset would looks in this way:

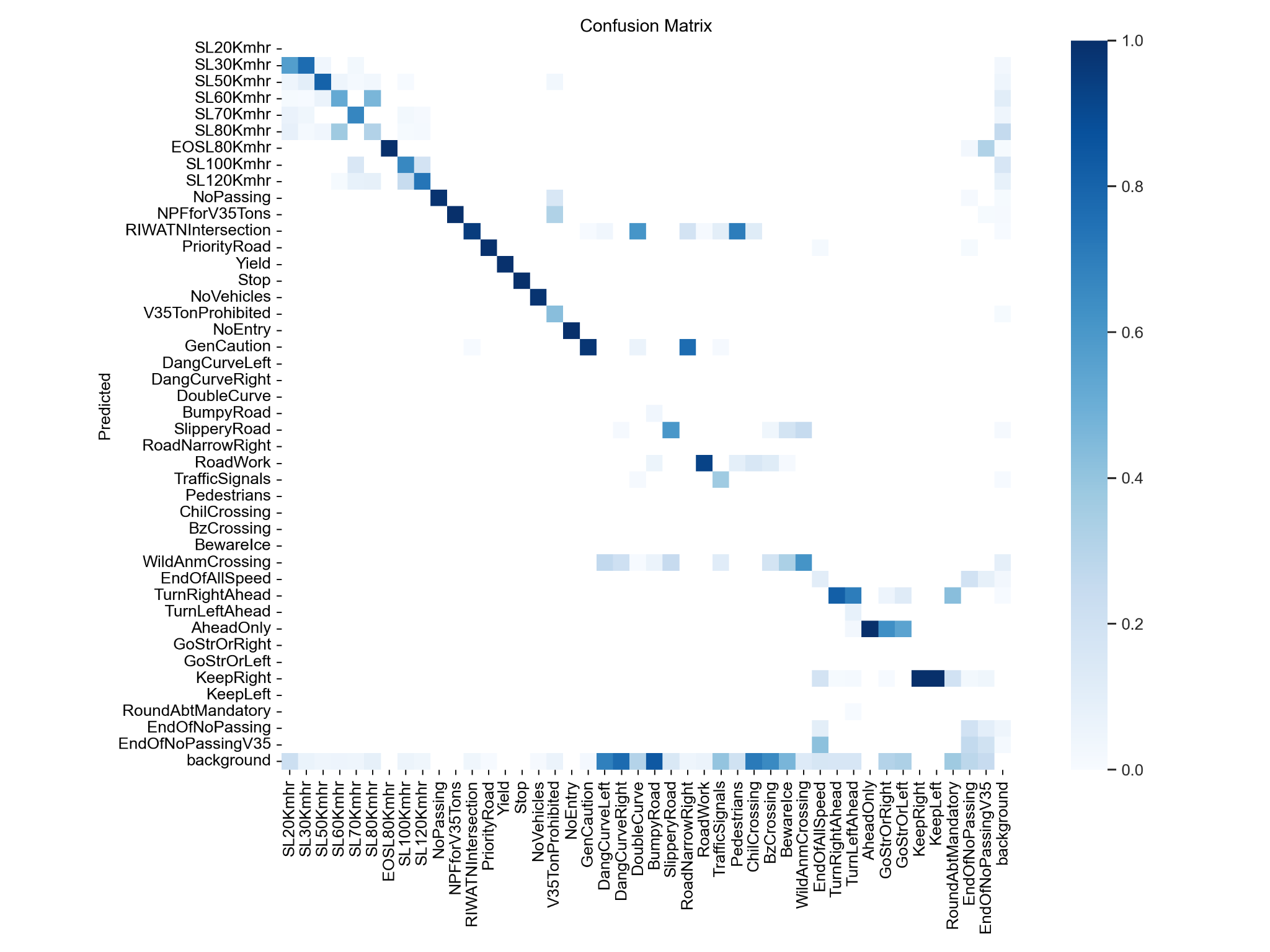


Fig 2: Confusion matrix of YOLO model

Here is an example of applying our trained model on the test set.



Fig 3: Apply model on test set

Model-2 was trained on CCTSDB (Changsha University of Science and Technology Chinese traffic sign detection benchmark) which contains more than 15 thousands images for training purposes. 由于GTSRB已经是10k级别的数据，为了是实现的多样性，我们从其中随机选取了大约3k张做thousand 级别的数据训练。

CCTSDB中的图片都是标注好的数据，所以需要自己手动划分训练集和测试集. 训练集中，我们按照8：2的比例进行了手动。由于CCTSDB中的数据也是全部保存在一个txt文件中，我们再次应用上述公式生成Image 对应的Label。

The batch size of the model is trained from 16 to 256 (n times of 16), however, we got the best accuracy when the batch size is 32.

Model-3 was trained on GTSDB (German Trafﬁc Sign Detection

Benchmark) is a single image detection dataset which contain 900 ppm images. 600 ppm image for train and 3000 ppm image for test. We also apply the YOLO model on this dataset. The images are initially released for a face recognition competition but later released as a public dataset.

通过对以上三个dataset 的应用，我们可以初步的认为我们的模型在交通标志识别领域比较有效，能够在一定程度上满足工业的挑战。

The proposed method is a one-stage method. All the experiments are performed in a GPU 2080 ti, I9-9900K, 32GB Ram CUDA 10, Pytorch 1.8.0 machine.

**Chapter 4 Model Analysis**

The model 将会被从mAP，AP，precision和recall四个方面进行评价。

The precision is a measure of metrics of quality，是我们预测为正确预测为正样本的数量除以预测为正样本的数量。Recall 则是查全率，表示我们正确预测为正确本的数量占所有正确正样本的数量。The precision and recall is defined with the following metrics:

Precision = TP/(TP+FP)

Recall = TP/(TP+FN)

Where TP 表示预测值和真实值一样，且都为正样本，TN表示预测值和真实值一样，且都为负样本，FP表示预测值为正样本，真实值是负样本，FN表示预测值为负样本，真实值是正样本。

但是这两种评价指标都只能在一定程度上反应模型的性能，并不能准确地表示模型，因此，我们又引入了 AP和mAP。PR-Curve值是precision和recall构成的的曲线，AP则是这两个值构成的曲线的线下面积。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**Chapter 5 Conclusion**

这次研究的主要目的是研究深度学习在交通标志中的应用，我们通过对数据集原来的进行一定的变换进行将合成图像与原始图像结合，来增强数据集并且提高深度学习模型的效果。

Future Research is aimed to deal with the dataset, such as we can use the GAN method to improve various kinds of images that are hard to be recognized and then train our model with them to improve the accuracy of the model. We can also improve the model by using Deep autoencoders which can help us detect traffic signs while leaving any other objects with only traffic signs.

**Reference**

[1] 尹宏鹏, 陈波, 柴毅, and 刘兆栋, "基于视觉的目标检测与跟踪综述," *自动化学报,* vol. 42, no. 10, pp. 1466-1489, 2016.

[2] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *science,* vol. 313, no. 5786, pp. 504-507, 2006.

[3] A. Liu, M. S. Khatun, H. Liu, and M. H. Miraz, "Lightweight Blockchain of Things (BCoT) Architecture for Enhanced Security: A Literature Review," in *2021 International Conference on Computing, Networking, Telecommunications & Engineering Sciences Applications (CoNTESA)*, 2021: IEEE, pp. 25-30.

[4] 卢宏涛 and 张秦川, "深度卷积神经网络在计算机视觉中的应用研究综述," *数据采集与处理,* vol. 31, no. 1, pp. 1-17, 2016.

[5] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580-587.