“三跨”输电线路智慧巡检中的车辆检测技术研究

沙洁韵

目录

CONTENTS

Part 1 研究意义

Part 2 工程车辆检测算法设计

Part 3 车流量统计方法设计

Part 4 软件平台设计与实现

Part 5 总结与展望

研究意义

1／ 研究意义

2／ 现存问题

3／ 主要工作

研究意义-三跨输电线路重要性

“三跨”输电线路：

跨越高速公路

跨越高速铁路

跨越重要输电通道

“三跨”输电线路因其架设位置的特殊性，与铁路、公路和特高压电力线等重要设施邻近或交叉，一旦出现安全问题，会对邻近的交通和电力基础设施造成破坏，影响居民生活和工业生产用电。

研究意义-车辆检测与流量统计必要性

- 重要输电通道——工程车检测，实现对可能存在的输电线路外力破坏进行预警的功能。
- 高速铁路——监测高铁运行状态，判断输电线路运行是否稳定
- 高速公路——统计车流量，为输电设备安全体系建设提供可靠的外部数据

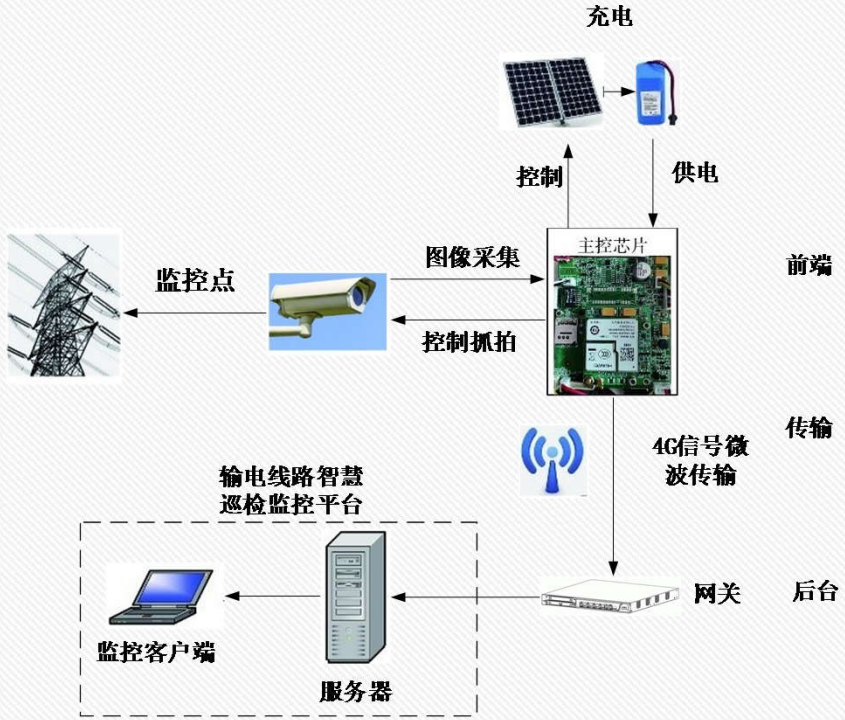
研究意义-智慧巡检

人工巡检

“三跨”输电线路架设范围广，人工巡检的方式效率低下地形情况复杂，天气情况变化莫测，人工巡检危险性大人工巡检成本高，耗费人力

智慧巡检

架设在杆塔上的摄像头实时采集视频图像传输给后端服务器，监测输电线路运行状态，采集外部交通数据，建立智慧电网系统



研究意义-不同方法的优劣

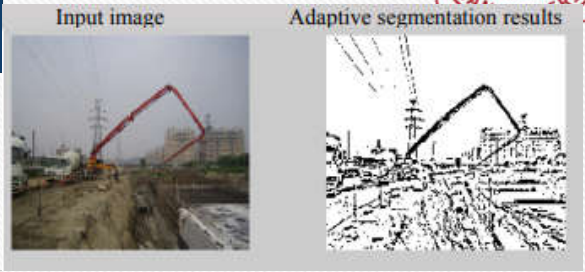
输电线路内车辆检测

传统图像处理方法

图像预处理，LOG算子、Sobel算子边缘检测

边缘检测算法对图像噪声、亮度敏感

不能直观的给出目标的位置和类别信息



机器学习的方法

提取HOG特征，使用SVM训练目标分类模型

手工设计目标特征，依赖于人为的先验因素，表达能力有限

不具有泛化性，不适用于复杂多变的环境

深度学习的方法

目标检测网络，R-CNN系列，YOLO系列

自动提取车辆深层特征

精确检测，鲁棒性强

研究意义-不同方法的优劣

车流量统计

传统运动目标检测方法

背景差分法，帧差法，光流法

不能准确检测高速和低速运动的物体，不能应对复杂背景，对光照亮度变化敏感

（a）道路 1 原始图像　（b）道路 1 帧差图

深度学习结合目标跟踪

深度学习检测车辆结合跟踪算法

抗背景噪声干扰能力强，车辆检测准确度高

跟踪算法确定车辆运动轨迹，流量统计准确度高

现存问题-深度学习检测方式存在的问题

一、数据集

卷积神经网络的训练本质上是学习大量的数据特征，输电线路内的各类工程车辆，较少有公开的数据集可以使用

二、误检漏检现象普遍

输电线路走廊多位于偏远地域，环境多变，背景复杂，工程车辆的颜色和形体常与背景混杂在一起难以识别，且监控摄像头架设位置较高较远，使画面中车辆的尺度变化较大，导致漏检。

三、算法实时性不高

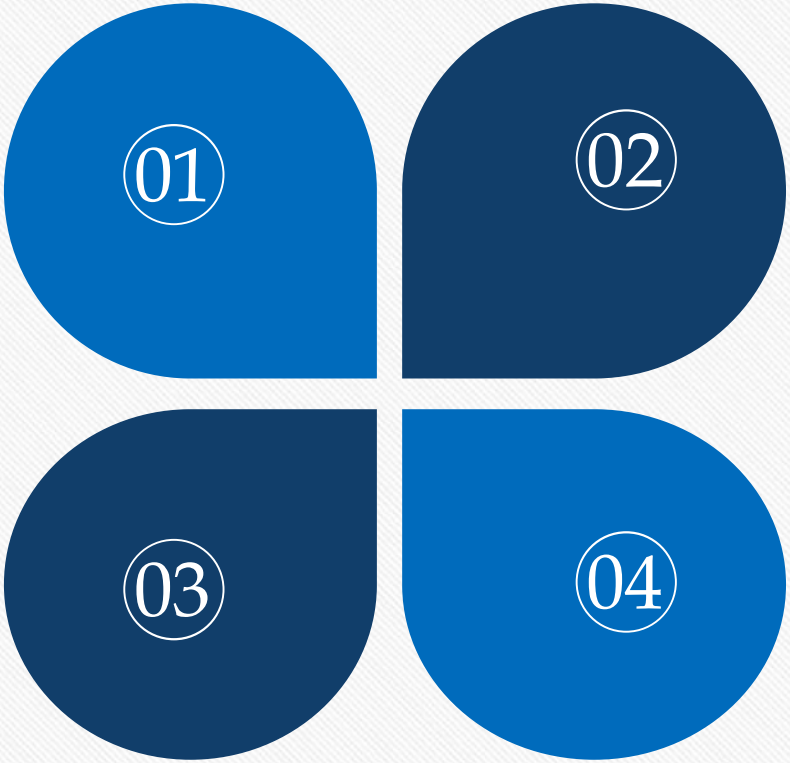
车辆检测模型参数量大，计算复杂度高，

车流量法整体耗时较长，不能达到实时统计的要求。

主要工作

一、数据集

完成输电线路图像的采集和车辆标注工作，建立适用于深度学习平台的车辆数据集并对图像进行增强处理



二、网络模型

适用于车辆检测的深度学习网络(YOLOv5目标检测网络)，并对检测算法进行优化提高检测准确度

三、流量统计

轻量化车辆检测模型，结合卡尔曼滤波跟踪算法确定车辆运动轨迹，提高车流量统计实时性

四、软件平台

基于Django框架搭建车辆检测和车流量统计软件平台，展示算法效果

工程车辆检测算法设计

1／ 建立数据集

2／ 网络模型搭建与优化

3／ 实验结果与分析

建立数据集-图像采集与标注

车辆图像采集

类别	塔吊	挖掘机	铲车	推土车	渣土车	水泥罐车	高速火车
标签	towercrane	excavator	digger	bulldozer	truck	cement	train

塔吊

渣土车 挖掘机

水泥罐车

推土车 铲车

高速火车

建立数据集-图像采集与标注

LabelImg标注

```
<path>F:\工程车分类\工程车图片\TEMP1\000023.jpg</path>
- <source>
    <database>Unknown</database>
  </source>
- <size>
    <width>1280</width>
    <height>720</height>
    <depth>3</depth>
  </size>
  <segmented>0</segmented>
- <object>
    <name>towercrane</name>
    <pose>Unspecified</pose>
    <truncated>0</truncated>
    <Difficult>0</Difficult>
  - <bndbox>
      <xmin>141</xmin>
      <ymin>143</ymin>
      <xmax>216</xmax>
      <ymax>304</ymax>
    </bndbox>
  </object>
- <object>
    <name>towercrane</name>
    <pose>Unspecified</pose>
    <truncated>1</truncated>
    <Difficult>0</Difficult>
  - <bndbox>
      <xmin>1</xmin>
      <ymin>124</ymin>
      <xmax>80</xmax>
      <ymax>304</ymax>
    </bndbox>
  </object>
</annotation>
```

训练图像路径、名称

图像大小

目标类别、在图像中的位置

建立数据集-图像增强必要性

存在的问题

① 图像整体亮度低，噪声多

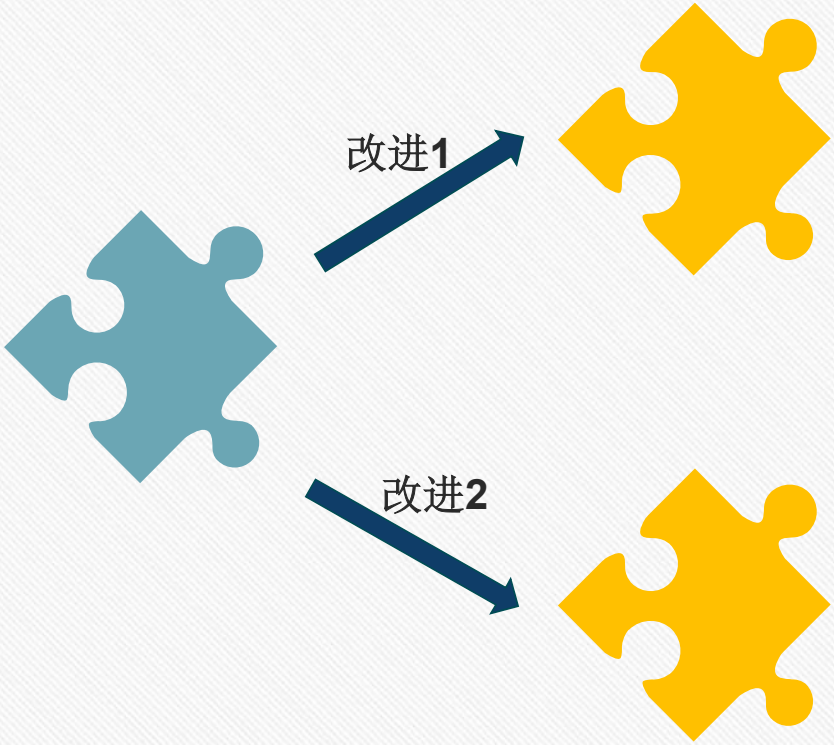
② 工程车辆颜色与形体与背景混杂

③需要对图像进行增强处理，提高车辆和背景的对比度

建立数据集-改进的直方图均衡化图像增强算法

传统的直方图均衡化算法

通过图像的灰度直方图确定一个映射函数，将像素值的分布从原来的相对集中状态变换为在各个像素值上均匀分布，均衡化操作涉及全局图像，容易导致部分区域对比度增强过大，成为噪点。一些感兴趣区域均衡化后会丢失必要的细节信息，



将原始图像分成多个矩形块，对于每个矩形块，分别统计其灰度直方图进行均衡化处理

改进的直方图均衡化算法

针对每个矩形块的灰度直方图，通过裁剪的方式使直方图的最大值低于某个设定的值，以此限制图像局部的对比度。同时为了控制直方图的总面积保持不变，将裁剪出的部分平均分配到各个灰度级。

建立数据集-改进的直方图均衡化图像增强算法演示图

(a)原图1

(b)传统直方图均衡化图像

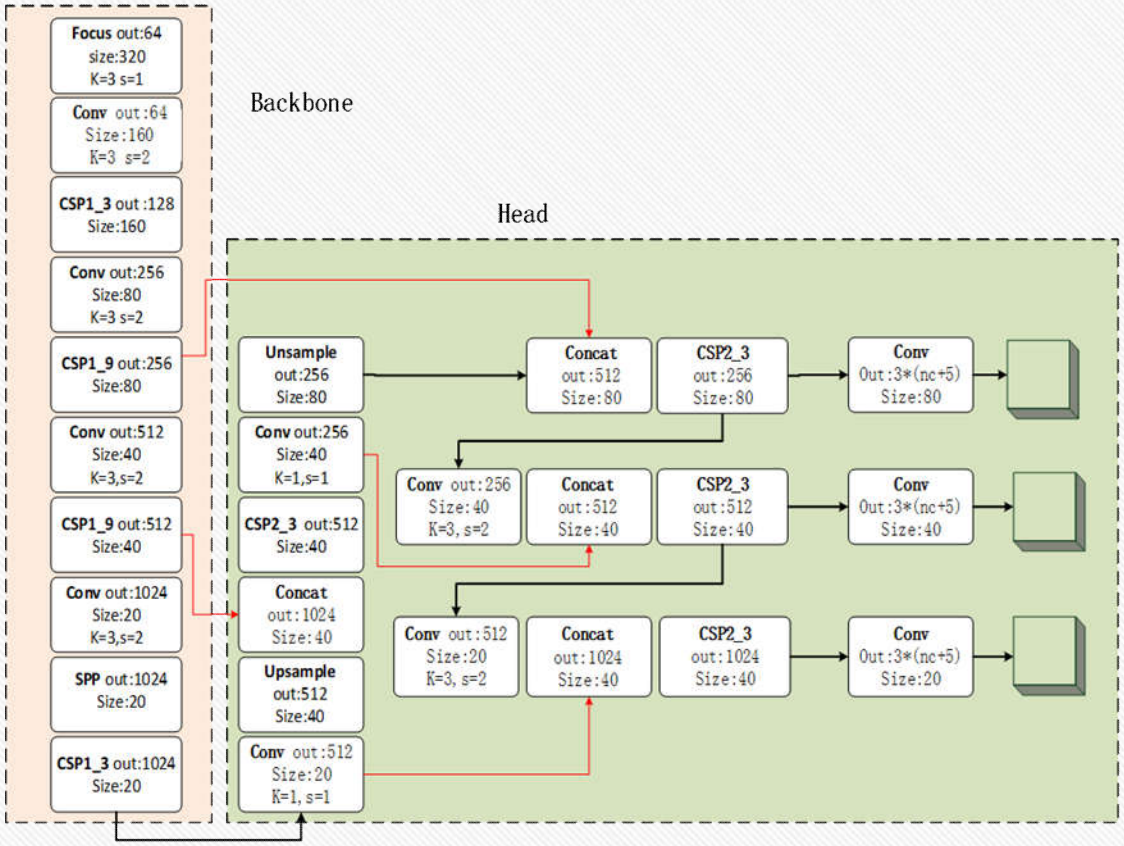
(c)改进的直方图均衡化图像

(a)原图2

(b)传统直方图均衡化图像

(c)改进的直方图均衡化图像

网络模型搭建与优化-YOLOv5网络模型



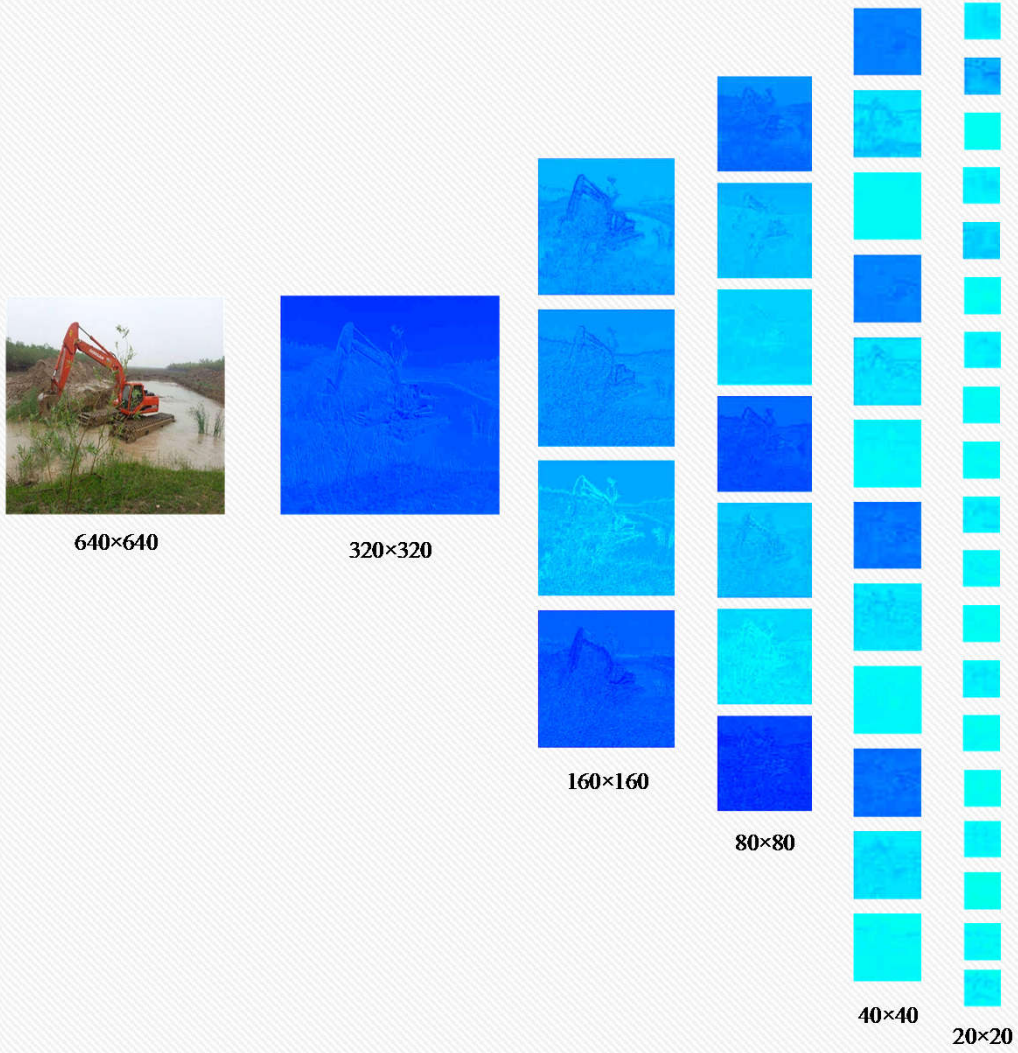
YOLOv5网络结构：

- 输入像素级切片操作，下采样保留细节信息
- 跨阶段局部网络结构，自动提取目标特征
- 空间金字塔池化结构
- 多尺度特征图构建

优点：

- 网络结构灵活性强，可自由调整深度和宽度
- 计算量少，内存成本低
- 最大程度保留了浅层映射图具有的车辆细节特征
- 端到端

网络模型搭建与优化-卷积层可视化



不同卷积层输出特征：

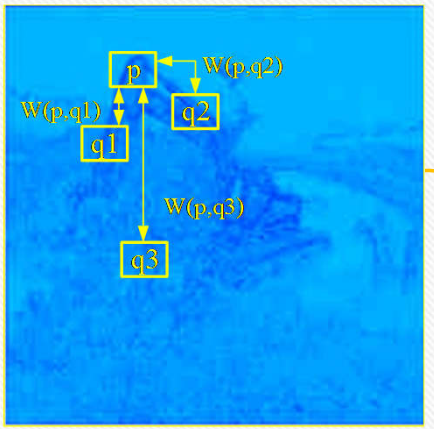
浅层卷积输出：640×640-160×160
得到小感受野，大分辨率的特征图，提取目标的轮廓纹理等几何特征，工程车辆的主要部件的形态可以与背景较好的区分，语义信息缺乏。

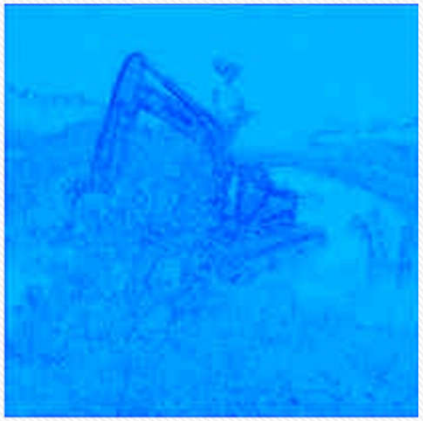
深层卷积输出：80×80-20×20
得到大感受野，小分辨率特征图，虽丢失车辆几何信息，但提取到更抽象，鲁棒性更强的语义信息。

网络模型搭建与优化-非局部均值

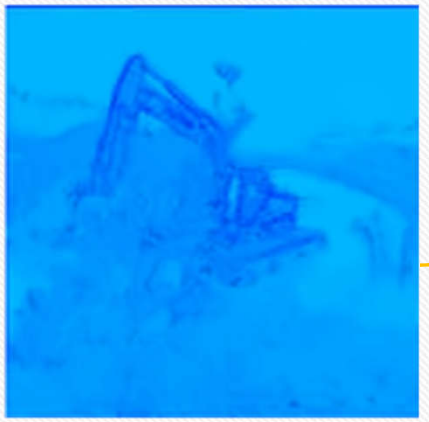
非局部均值滤波算法

图像当前像素点的估计值由与它具有相似邻域结构的像素点的值加权平均得到。在具体操作中，为了减少逐像素计算的复杂度，采用块(block)思想，计算图像中每个block的相关性





Non Local Means



作用于卷积特征图：

针对特征图中的每个特征点，计算与其他所有特征点的远距离依赖关系并进行加权平均计算，关联全局特征信息之后，特征图中的背景噪声点被有效的削弱，突出了车辆的的特征信息，有利于后续对目标的识别和定位。

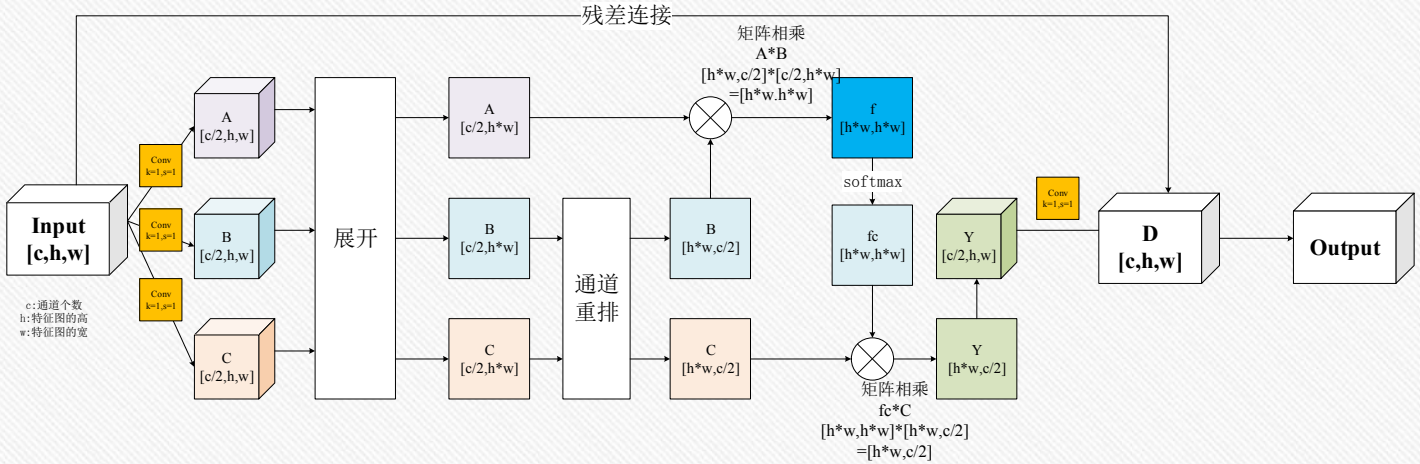
网络模型搭建与优化-非局部模块

非局部操作：

$$y_i = \frac{1}{C(x)} \sum_{\forall j} f(x_i, x_j) g(x_j) \qquad g(x_j) = W_g x_j$$

特征图中每个特征点融合全局的信息分布

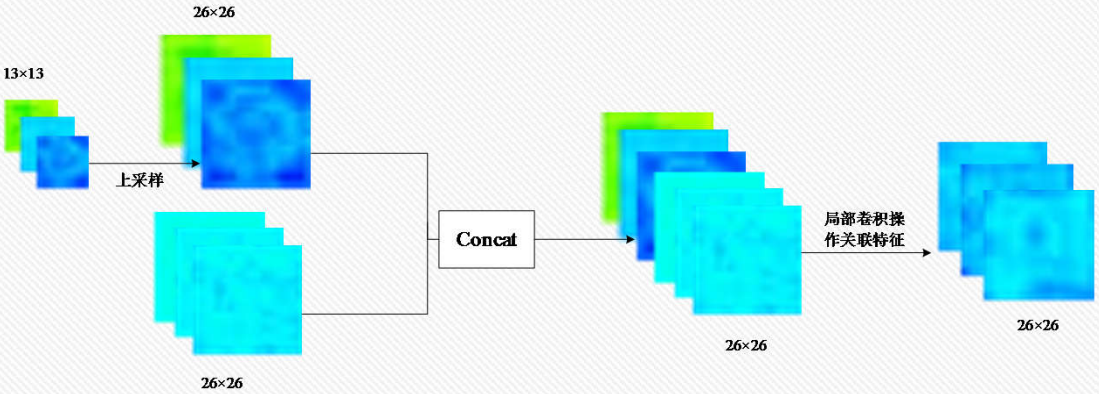
非局部模块:



通过1×1的卷积提取全局特征，特征张量进行矩阵相乘的操作计算出一个特征点与全局其他特征点的相似度，并对特征响应值进行了加权平均操作，softmax函数对加权平均后的结果进行了归一化。这一过程充分融合了输入特征图的全局信息，强化了对检测任务有利的特征点，削弱了噪声点对检测的负面影响，扩大了输出特征图的感受野，有利于后续对多尺度目标的识别和定位。

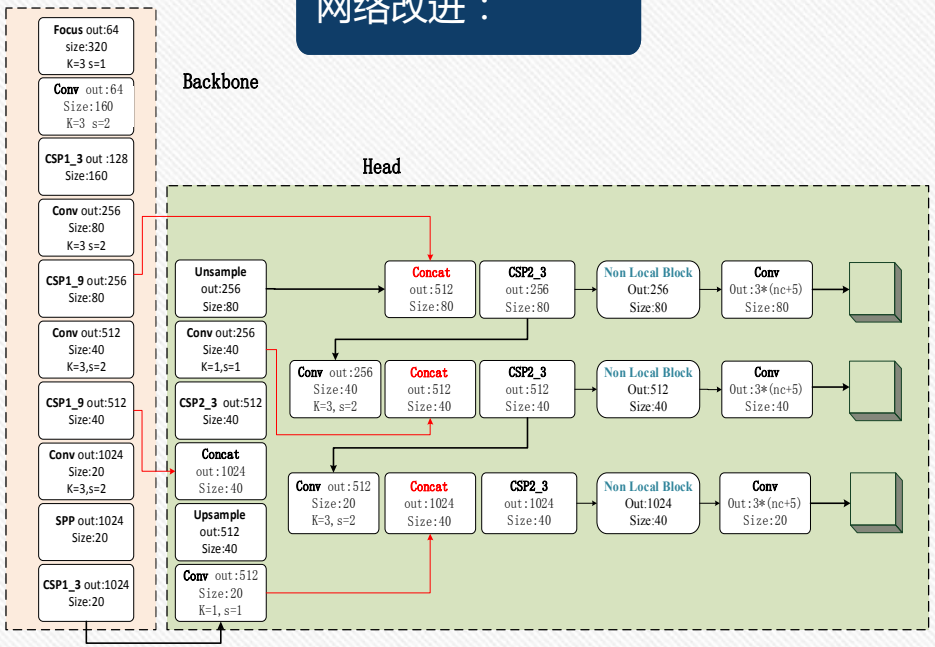
网络模型搭建与优化-网络改进

多尺度融合操作：



为了融合网络不同卷积层中的输出特征信息，通过concat操作将不同卷积层的输出分辨率相同的特征图进行通道拼接，但通道层面的特征图拼接并没有对特征图本身的信息进行融合

网络改进：



将非局部模块引入CSP2模块之后，对多尺度特征融合后的输出特征图中的每个点，计算与其他特征点的相似度并进行加权平均的操作，强化对检测任务有利的特征点，减弱噪声点的分布，使最终输出的特征图的感受野扩大至全局，获取表达能力更强的语义信息，增强对多尺度目标的检测能力。

实验结果与分析-实验数据及环境

输电线路内车辆数据集：

车辆种类	训练集标签数	验证集标签数	测试集标签数
塔吊 towercrane	1206	229	554
挖掘机 excavator	3287	510	919
铲车 digger	1017	165	526
推土车 bulldozer	978	139	390
渣土车 truck	1071	212	515
水泥罐车 cement	1032	190	499
高速火车 train	373	74	148
总数	8964	1519	3551

实验环境配置：

CPU： Intel(R) Core(TM) i3-9100F CP@ 3.60GHz

GPU： GeForce RTX 2080 Ti

运行内存： 31G

软件平台： JetBrains Pycharm

深度学习框架： PyTorch 1.7.1

编程语言： Python 3.7.4

训练参数设置：

初始学习率learning rate： 0.01

优化器： SGD

动量momentum： 0.937

图像缩放尺寸： 640×640

批训练数量batch-size： 16

训练轮次 epoch： 300

实验结果与分析-评价指标

	正样本	负样本
网络检测为正样本	True Positive	False Positive
网络检测为负样本	False Negative	True Negative

召回率： $$Recall = \frac{TruePositive}{TrurPositive + FalseNegative} \times 100\%$$

精确率： $$Precision = \frac{TruePositive}{TrurPositive + FalsePositive} \times 100\%$$

平均精确率： $$AP = \sum_{k=1}^{N} P(k)\Delta r(k) = \int_0^1 p(r)dr$$

实验结果与分析-实验结果

(a)YOLOv3

(b)YOLOv5

(c)本文方法

(a)YOLOv3

(b)YOLOv5

(c)本文方法

实验结果与分析-实验结果

(a)YOLOv3

(b)YOLOv5

(c)本文方法

(a)YOLOv3

(b)YOLOv5

(c)本文方法

实验结果与分析-实验结果

(a)YOLOv3

(b)YOLOv5

(c)本文方法

(a)YOLOv3

(b)YOLOv5

(c)本文方法

实验结果与分析-实验结果

检测结果对比：

	mAP	塔吊	挖掘机	铲车	推土车	渣土车	水泥罐车	高速火车
YOLOv3	90.3	80.2	92.4	94.5	96.1	86.6	86.6	95.6
YOLOv5	92.3	82.6	94.9	95.5	97.9	90.6	90.2	94.3
本文方法	93.5	85.8	95.3	96.1	98.2	92.5	90.8	96.0

结果分析：

与YOLOv3网络相比，选择性能更好的YOLOv5框架是提高车辆检测性能的基础。针对输电线路这一应用场景，使用改进的直方图均衡化算法对数据集进行图像增强处理，可以提高图像整体亮度、工程车辆和背景的对比度，强化车辆的有效特征。结合引入非局部模块的改进的车辆检测网络框架，基于增强数据集训练的检测模型能提高工程车辆检测的准确度，减少漏检情况，验证了本文针对“三跨”输电线路内工程车辆检测算法改进思路的有效性。

车流量统计方法设计

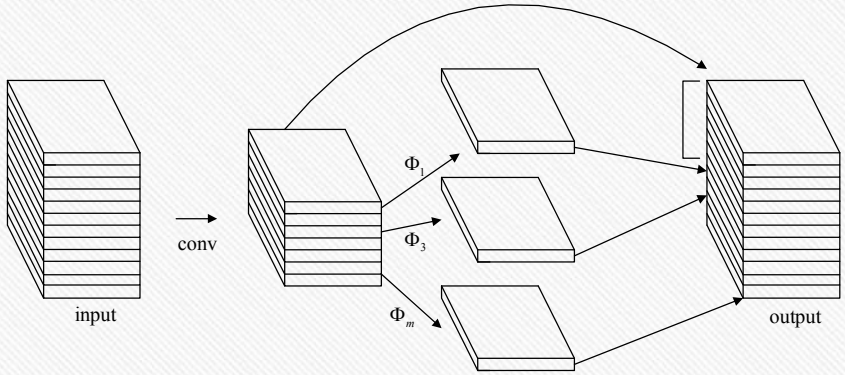
1／ 车辆检测算法轻量化改进

2／ 多目标车辆跟踪

3／ 实验结果与分析

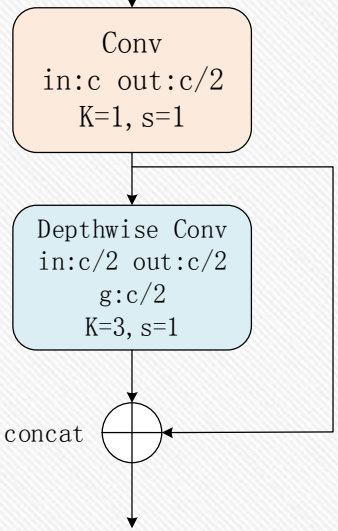
车辆检测算法轻量化改进-轻量化模块Ghost

幻影特征图生成：



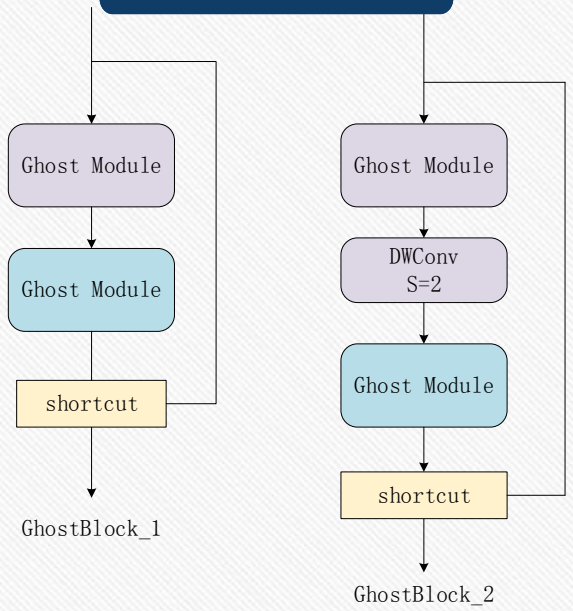
基于少量原始特征图，采用简单的线性运算生成有效的幻影特征图

Ghost Module：

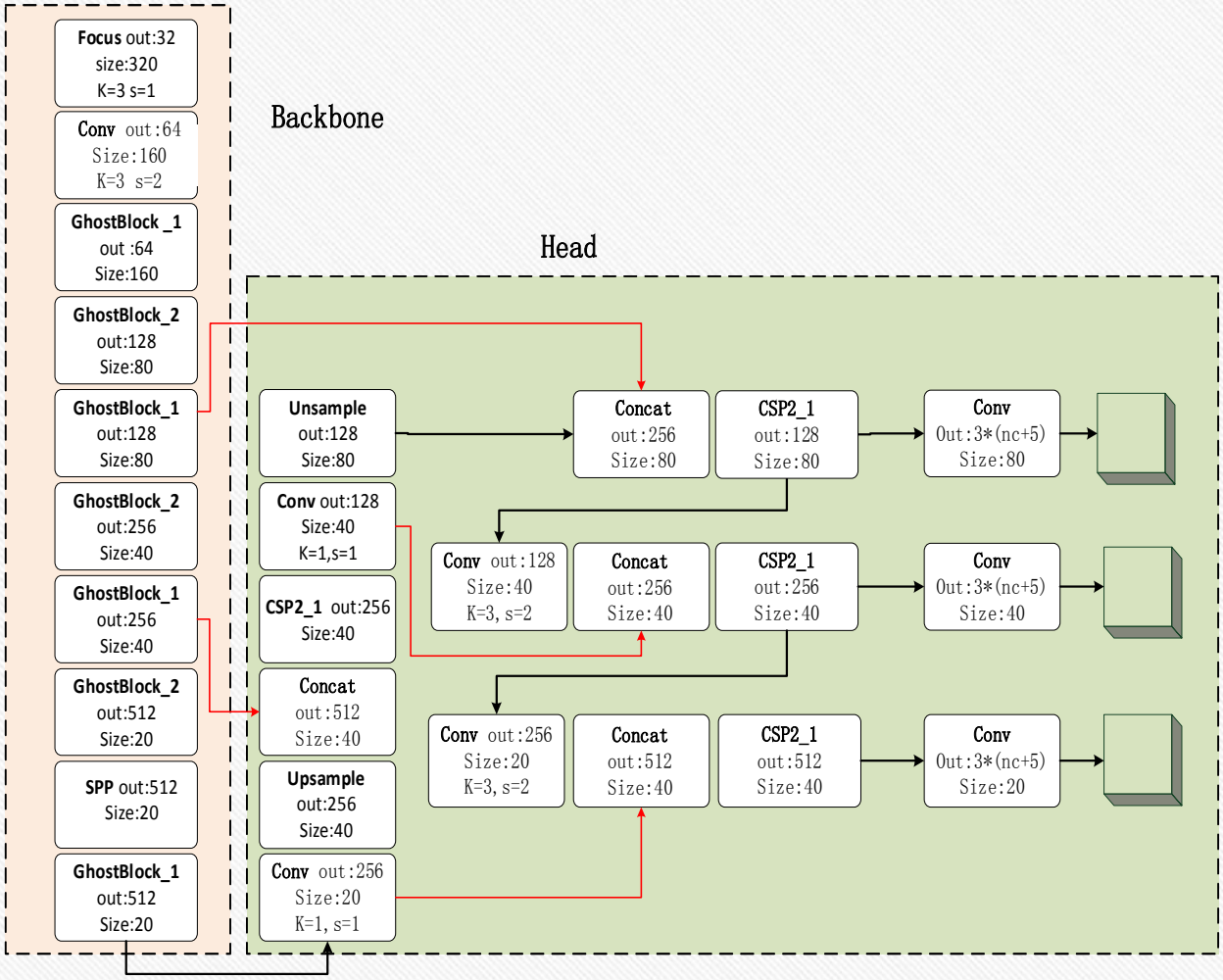


输入模块的特征图通道数为c ，先使用尺度1×1的c/2个卷积核提取特征，生成通道数为c/2的原始特征图，基于第一阶段的原始特征图，使用3×3的c/2个卷积核进行深度可分离卷积(Depthwise Convolution，DWConv)操作,并与第一阶段的输出特征图进行通道拼接

Ghost Block：



车辆检测算法轻量化改进-轻量化YOLOv5网络框架



- 在骨干网络模块(Backbone)部分，使用GhostBlock_1和GhostBlock_2替换原本的特征提取模块CSP_1以及后四个步长为2的下采样操作。
- 将特征检测层Head中CSP_2模块的普通卷积子模块个数减为1，并将网络中所有模块的输出特征图通道数减半，进一步简化模型结构。
- 将轻量化后的YOLOv5网络框架命名为Ghost-YOLOv5

多目标车辆跟踪-卡尔曼滤波算法

当前帧中车辆的状态，即位置(position)和速度(velocity)满足高斯分布，表示为：

$$\hat{x}_t = \begin{bmatrix} p_t \\ v_t \end{bmatrix}$$

令车辆的加速度为 u_t 和上一帧中此车辆的状态关系可以表示为

$$p_t = p_{t-1} + \Delta t \cdot v_{t-1} + \frac{1}{2} u_t \times \Delta t^2$$

$$v_t = v_{t-1} + u_t \times \Delta t$$

转换为矩阵形式为

$$\hat{x}_t = \begin{bmatrix} p_t \\ v_t \end{bmatrix} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} p_{t-1} \\ v_{t-1} \end{bmatrix} + \begin{bmatrix} \Delta t^2 / 2 \\ \Delta t \end{bmatrix} u_t \longrightarrow \hat{x}_t^- = F_t \hat{x}_{t-1} + B_t u_t$$

状态预测方程1

引入系统的不确定性和噪声

$$P_t^- = F_t P_{t-1} F_t^T + Q_t$$

状态预测方程2

多目标车辆跟踪-卡尔曼滤波算法

引入观测的不确定性矩阵 R,观测矩阵为H 观测值和真实值的关系为

$$Y_t = H\hat{x}_t + \mathrm{R}$$

观测值和真实值为对状态预测方程进行修正

$$\hat{x}_t = \hat{x}_t^- + K_t(Y_t - H\hat{x}_t^-)$$

$$K_t = P_t^- H^T \left(HP_t^- H^T + R\right)^{-1}$$

$$P_t = \left(1 - K_t H\right) P_t^-$$

修正方程

使用状态预测方程1和2预测某一帧中的车辆在下一帧中的状态，结合修正方程可以得到对目标车辆的最优估计值

多目标车辆跟踪-预测与检测结果匹配

(a)车流量视频第i帧

(b)车辆检测结果

(c)车辆状态预测

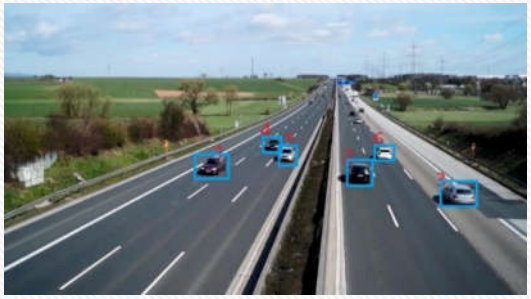
检测结果和预测结果计算交并比

	1	2	3	4	5	6
1	[[0.623	0.000	0.000	0.000	0.000	0.000]
2	[0.000	0.006	0.461	0.000	0.000	0.000]
3	[0.000	0.753	0.031	0.000	0.000	0.000]
4	[0.000	0.000	0.000	0.000	0.000	0.678]
5	[0.000	0.000	0.000	0.000	0.469	0.000]
6	[0.000	0.000	0.000	0.718	0.000	0.000]
7	[0.000	0.000	0.000	0.000	0.000	0.000]]

(d)交并比矩阵

```
[[1 1]
 [2 3]
 [3 2]
 [4 6]
 [5 5]
 [6 4]]
```

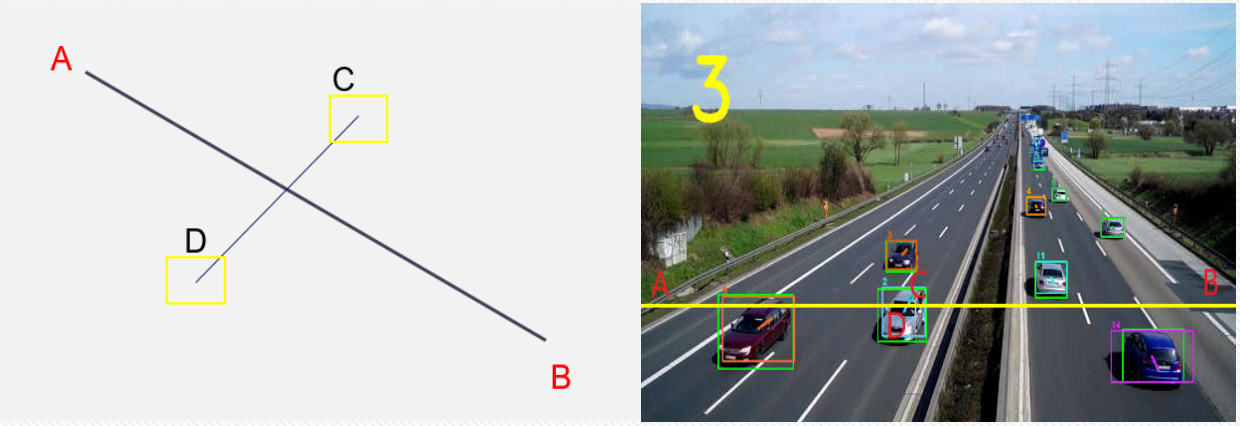
(e)匹配结果



(f)位置的最优估计值

多目标车辆跟踪-车流量统计实现

设置虚拟计数线：



虚拟计数线：AB

车辆运动轨迹：CD

$$(AB \times AC) \cdot (AB \times AD) \le 0$$

$$(CD \times CA) \cdot (CD \times CB) \le 0$$

车辆通过虚拟计数线

- 保存已经计数过的车辆编号， 避免重复计数
- 设置两条间隔5个像素点的虚拟计数线，减少计数遗漏
- 通过车辆位置中心的变化趋势，区分左右车道的车辆，分别计数

实验结果与分析-数据集与模型对比

高速公路数据集：

车辆种类	训练集标签数	验证集标签数	测试集标签数
卡车(truck)	20076	4234	8623
公共汽车(bus)	2834	666	1601
小汽车(car)	15066	3709	9168
总数	37976	8609	19392

高速公路车辆检测模型对比：

模型	mAP	Parameter	GFLOPS
YOLOv3	91.1	61.52×10^6	154.9
YOLOv5	93.0	47.38×10^6	116
Ghost-YOLOv5	91.9	4.54×10^6	10.5

实验结果与分析-车辆检测效果对比

(a)YOLOv3

(b)YOLOv5

(c)Ghost-YOLOv5

(a)YOLOv3

(b)YOLOv5

(c)Ghost-YOLOv5

实验结果与分析-车辆检测效果对比

(a)YOLOv3

(b)YOLOv5

(c)Ghost-YOLOv5

(a)YOLOv3

(b)YOLOv5

(c)Ghost-YOLOv5

实验结果与分析-车流量统计实验结果

视频序列	参数	真实数值	YOLOv5 统计结果	Ghost-YOLOv5 统计结果
Video_1	分辨率	1280*720		
	帧率	30FPS	25.13FPS	33.78FPS
	左车道	68	68(100%)	67(98.53%)
	右车道	82	82(100%)	82(100%)
Video_2	分辨率	1280*720		
	帧率	30FPS	24.88FPS	33.18FPS
	左车道	59	59(100%)	59(100%)
	右车道	48	48(100%)	48(100%)
Video_3	分辨率	1280*720		
	帧率	30FPS	25.98FPS	32.89fps
	左车道	84	83(98.81%)	84(100%)
	右车道	68	68(100%)	67(98.53%)

对比分析：

算法加载分辨率为1280×720，帧率为30FPS的视频进行车流量统计，使用轻量化车辆检测模型Ghost-YOLOv5与使用使用YOLOv5检测模型相比，车流量统计帧率从平均25.33FPS提高至33.28FPS，达到了实时统计的效果。

实验结果与分析-车流量统计实验结果

视频序列	参数	真实数值	YOLOv5 统计结果	Ghost- YOLOv5 统计结果
Video_4	分辨率	1920*1080		
	帧率	25FPS	16.44FPS	22.41FPS
	左车道	154	153(99.35%)	151(98.05%)
	右车道	135	135(100%)	134(99.25%)
Video_5	分辨率	1920*1080		
	帧率	25FPS	16.58FPS	22.45FPS
	左车道	134	134(100%)	134(100%)
	右车道	134	134(100%)	134(100%)
Video_6	分辨率	1920*1080		
	帧率	25FPS	16.53FPS	22.43FPS
	左车道	134	133(99.25%)	133(99.25%)
	右车道	136	135(99.26%)	135(99.26%)

对比分析：

加载分辨率1920×1080，帧率为25FPS的视频进行车流量统计，车流量统计帧率从16.52FPS提高至22.43FPS，基本达到实时统计的效果。

总结：

Ghost-YOLOv5与YOLOv5相比，由于模型轻量化过程中参数量和特征提取次数的减少，在部分视频序列上统计准确性略有下降，但仍保持了车流量统计98%以上的高准确度，实验结果验证了本问提出的对YOLOv5轻量化改进思路的有效性以及车流量统计方法的可行性。

实验结果与分析-车流量统计实验结果

(a) 第1帧

(b) 第1329帧

(c) 第1886帧

(d) 第2295帧

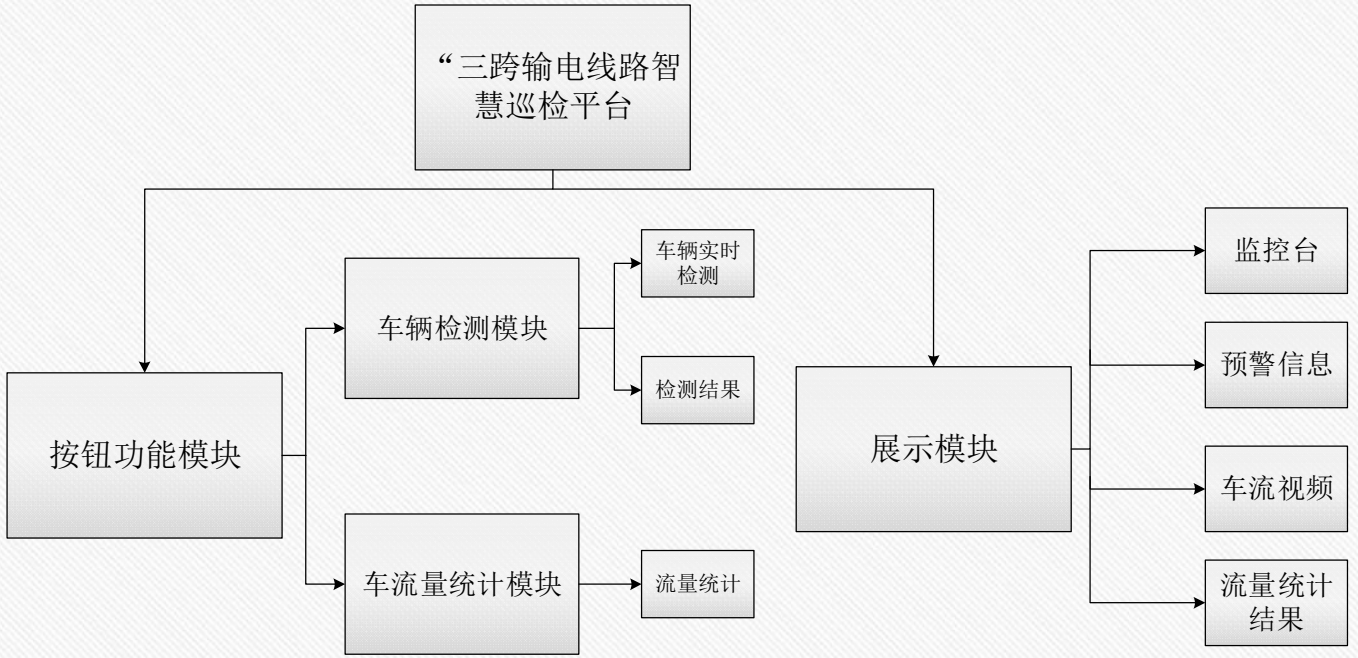
软件平台设计与实现

1／ 软件整体架构

2／ 功能展示

软件整体架构

本软件平台的开发过程在Linux系统下完成，主要分为界面开发和算法集成两个部分。界面开发部分使用Python语言，基于Web应用框架Django进行界面搭建。算法实现部分采用的编程语言同样为Python语言，开发平台为JetBrains PyCharm。



功能展示

车辆实时检测：

检测结果：

功能展示

车流量统计：

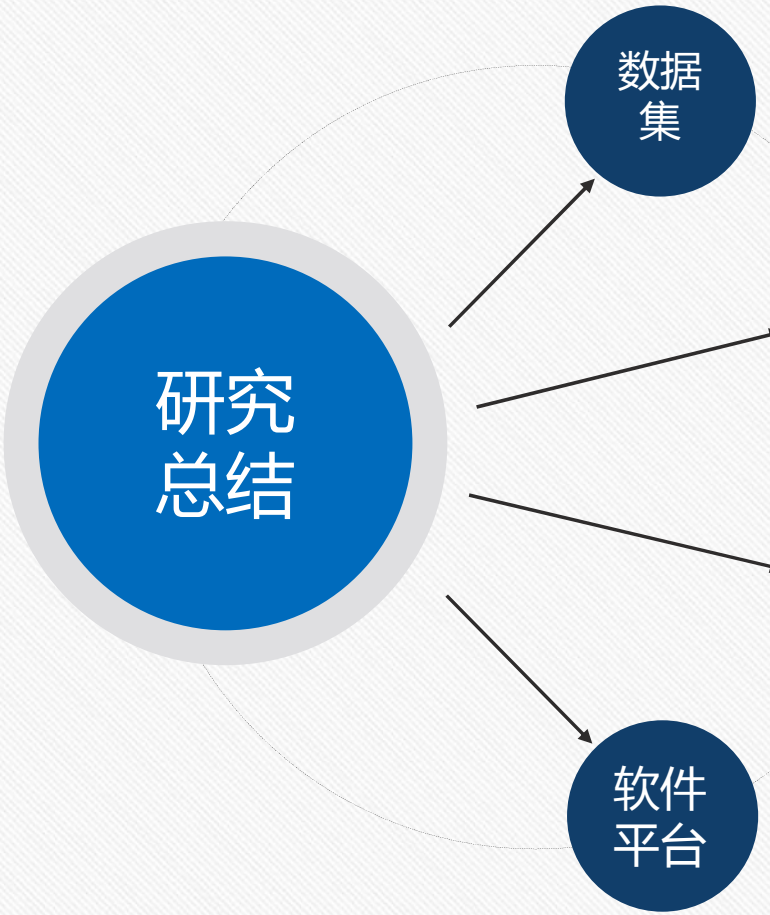
“三跨”输电线路智慧巡检平台
110kV 邹黄线58号
67
49

“三跨”输电线路智慧巡检平台
110kV 扬潭线2号
64
76

总结与展望

1／ 总结与展望

总结与展望-研究总结



自建了包含工程车和高速火车七种车辆类别的数据集，使用改进的直方图均衡化算法对图像进行增强处理，提高图像整体亮度和对比度，增强目标有效的细节特征。

采用一阶段目标检测网络YOLOv5搭建车辆检测框架，引入非局部模块，关联全局特征信息，强化对检测任务有利的特征点，抑制背景噪声等不利因素，提高了输电线路内多尺度工程车辆检测的准确度。

对YOLOv5网络框架进行轻量化改进，结合卡尔曼滤波算法实现了多目标车辆的跟踪，保证准确性的前提下提高了车流量统计的实时性。

采用B/S架构，基于Django应用框架搭建“三跨”输电线路内车辆检测与车流量统计软件平台，直观展示车辆检测和车流量统计的效果。

总结与展望-展望

一、

“三跨”输电线路内车辆图像采集存在困难，本文的数据集规模仍然有限，为了进一步提高检测模型的性能，后期可以考虑使用无人机等方式采集更多输电线路内的车辆图像，增强模型对各类工程车辆和高速火车特征的拟合效果。

二、

近年来有学者提出了不使用特征区域，而是采用关键点估计找到目标中心点的网络框架，后期可以将其与一阶段目标检测网络进行性能对比，研究YOLOv5的进一步优化方向。

三、

实验表明轻量化改进后的车流量算法在多个视频上统计的准确率仍有98%以上，但检测模型参数的减少仍带来了一定的精度下降，后期可以研究在轻量化改进的基础上，不引入其他复杂度高的操作，进一步提高车流量统计的准确度。