基于视频流的火情检测技术研究

宁 阳

目录

Contents

PART 01
论文背景与意义

PART 02
火情检测网络

PART 03
加速检测算法

PART 04
火情分析

PART 05
平台搭建与测试

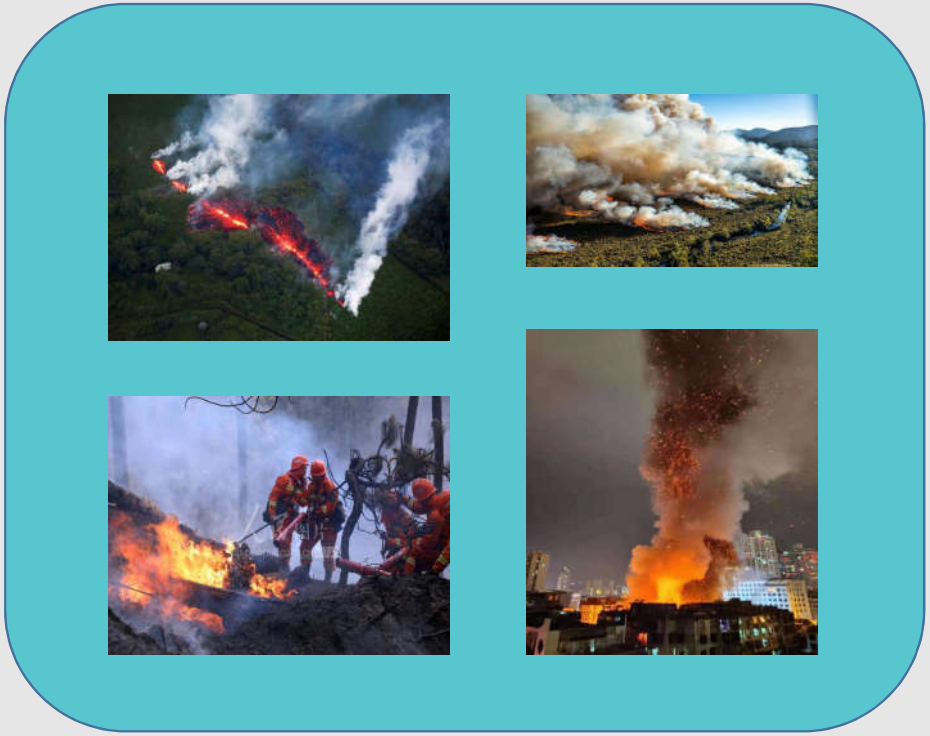
PART 06
总结与展望

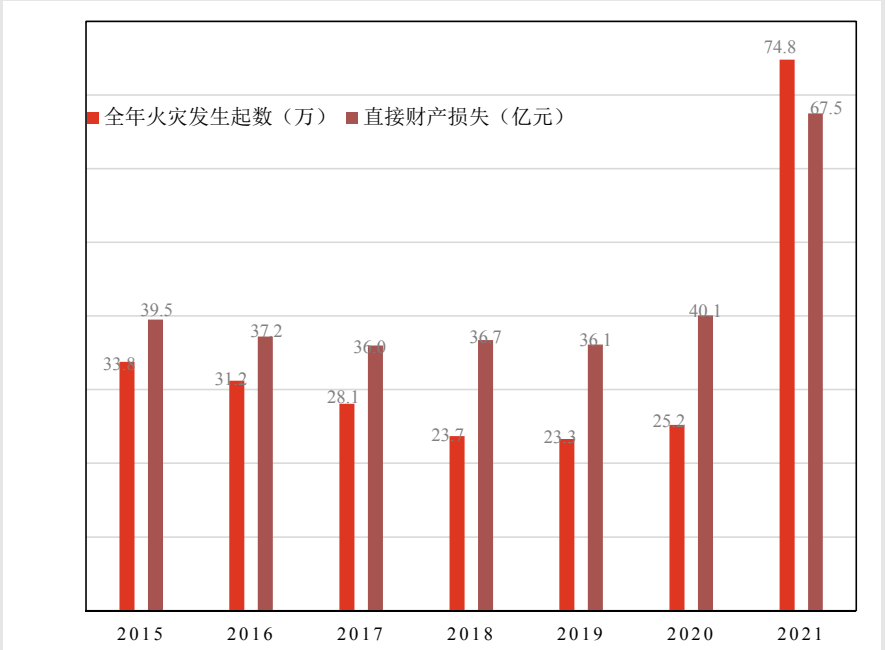
第一部分

论文背景与意义

- 研究背景
- 研究现状
- 现存问题
- 主要工作

1.1 研究背景




全年火灾发生起数（万）
直接财产损失（亿元）
74.8
67.5
39.5
37.2
36.7
36.1
40.1
33.9
31.2
28.1
23.7
23.3
25.2
2015
2016
2017
2018
2019
2020
2021

1.2 研究现状

（1）感光器型传感器：主要检测可燃物燃烧发出的光辐射，分为紫外线和红外线两种检测型号，通常用于发生可能爆炸情况的工厂仓库等环境中。若其受到其他如阳光长期照射时，容易发生误报。

（2）感烟型传感器：针对火情发生早期时的烟雾颗粒进行检测，是目前使用最广泛的类型，缺点是无法应对无烟火情。

（3）感温型传感器：主要是利用热敏元器件来探测火情，将温度变化转化为电信号进行报警，不过针对不同的使用场景需要调整检测阈值。

（4）气体型传感器：主要针对燃烧过程中产生的气体进行检测，包括一氧化碳和二氧化碳等气体的浓度变化，燃烧物气体等。但过高的生产成本导致了其难以被大规模应用。

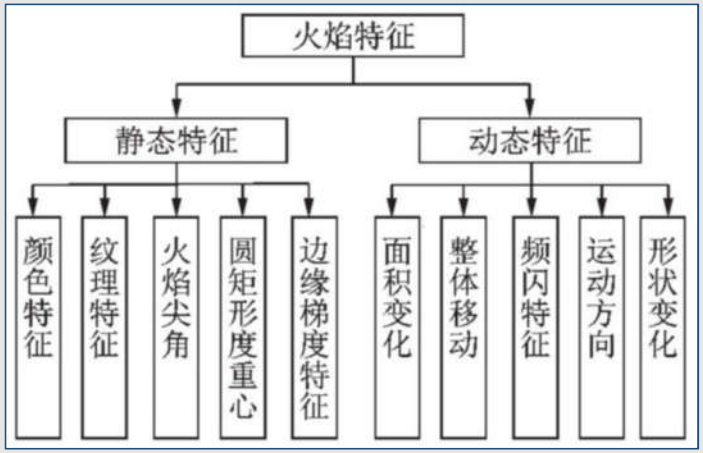
1.2 研究现状

传统图像检测

*Gubbi等人*提出通过小波变换来分解待测视频中单帧图像的方差、均值、梯度等特征的融合特征来描述烟雾，然后输入到SVM中进行分类。

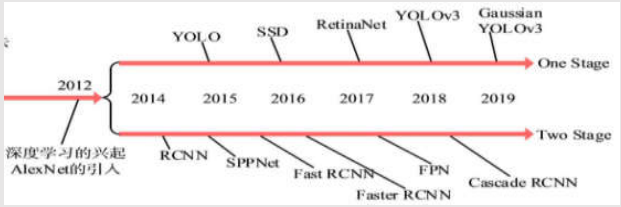
*Toreym等人*提出利用双层二维小波变换提取高频能量的大小来判断是否有烟雾

*王欣刚等人*利用运动目标检测获取运动区域然后通过烟雾产生时背景边缘发生模糊从而导致在小波频域上高频分量下降的特点进行检测，在测试集内达到85%的准确率。



1.2 研究现状

深度学习方法



基于候选区域

R-CNN：卷积网络提取特征，耗时长

SPP-Net：多尺度卷积特征

Fast R-CNN：用ROI pooling层提取特征

Faster R-CNN：提出区域生成网络

Two-stage

One-stage

YOLO：将图像划分为网格单元

YOLOV2：聚类产生锚框

YOLOV3：借鉴残差学习，进行多尺度检测

YOLOV4：马赛克数据增强，Mish激活函数等优化方法

SSD：生成多尺度的锚框对边界框空间离散化

DSSD：ResNet101为骨干网添加反卷积层

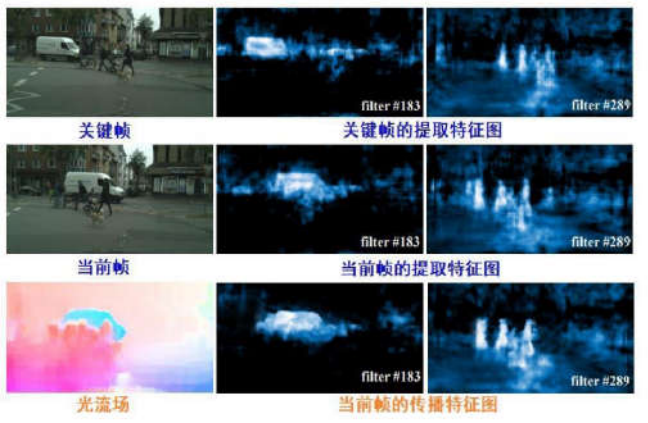
FSSD：借鉴FPN特征融合思想

ASSD：融合注意力机制

基于回归

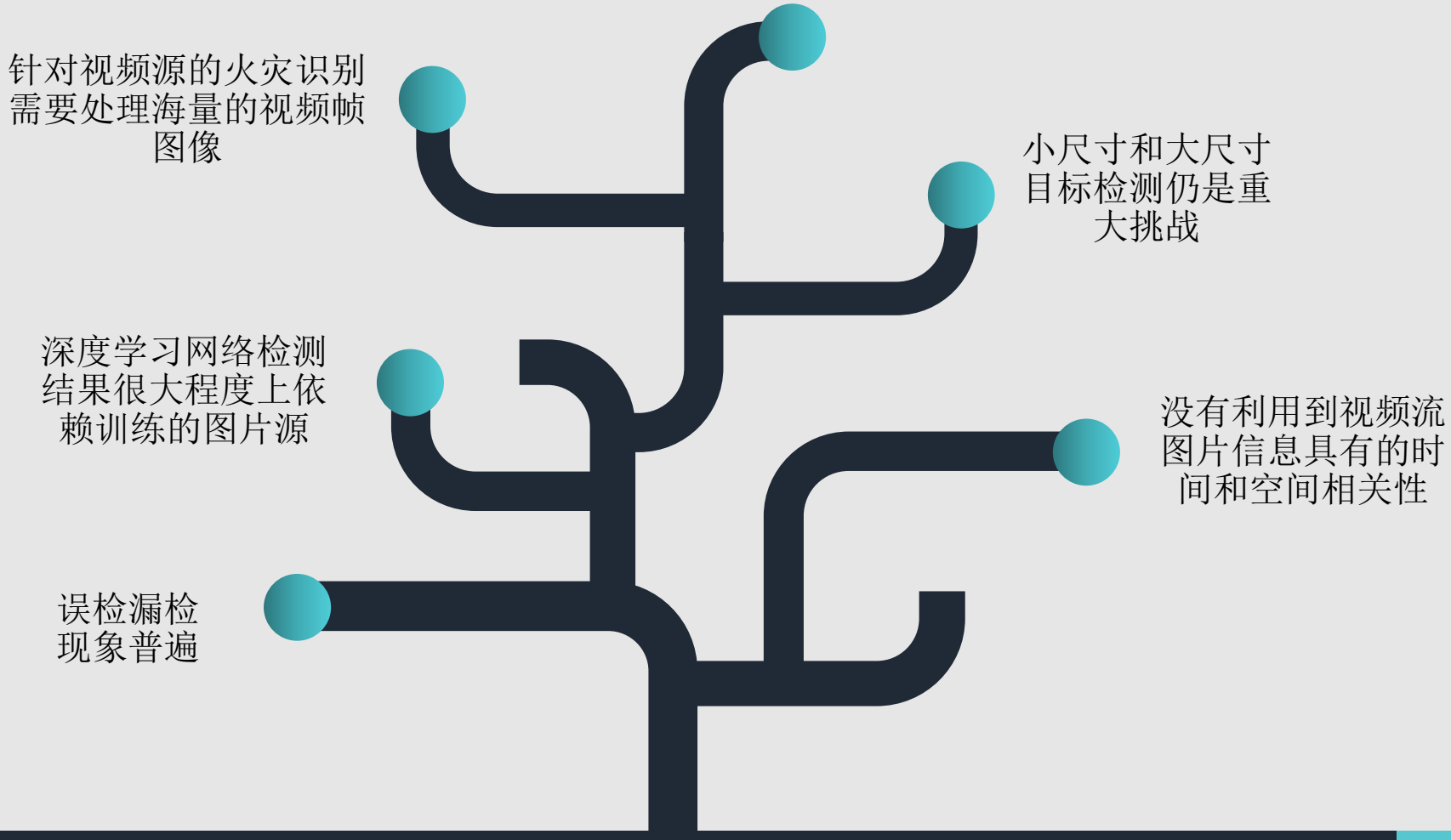
1.2 研究现状

视频目标检测



- DFF算法：把关键帧送入到深度卷积网络中提取特征，而对于非关键帧，算出与关键帧的光流，再用光流把关键帧的特征通过光流信息传播到当前帧。
- FGFA算法：通过将相邻帧在运动路径上的特征聚合到当前帧已有的提取特征中，首先在视频帧级别上比较充分地利用了时序信息，试图通过相邻帧的特征对当前帧持续聚合加强，从而提高视频目标检测的精度。

1.3 现存问题


针对视频源的火灾识别需要处理海量的视频帧图像
小尺寸和大尺寸目标检测仍是重大挑战
深度学习网络检测结果很大程度上依赖训练的图片源
没有利用到视频流图片信息具有的时间和空间相关性
误检漏检现象普遍

1.4 主要工作 | 研究目的

本研究为**面向视频流**的火情目标检测。旨在针对如居民楼，街道，仓库，厂区，输电线路，森林等一些复杂场景下的检测需求，设计一个完整的在线火情监控系统，采用深度学习的算法，搭建起基于视频流的检测架构，根据准确率，检测速度和可移植性等因素评价深度学习网络模型，最终可部署于监控地或借助已有摄像头进行实时准确的在线监控。该系统另包括后端浏览器**在线监控**查看、火情发生**预警信息**推送、**火情实时分析**等功能，以帮助消防人员进一步灭火或灾后分析等工作。

1.4 主要工作 | 已完成工作

数据集

建立起适合于CNN网络学习、训练的烟火图像数据集，包括图片的获取、处理与标注等。

网络优化

搭建目标检测网络模型，包括网络模型的选择、针对于特定的烟火目标而进行相关的网络改进等内容，以提高检测精度。

视频加速

根据目前视频目标检测算法存在的问题和弊端，提出一种MFF双网络特征融合算法，以提升检测速度。

火情分析

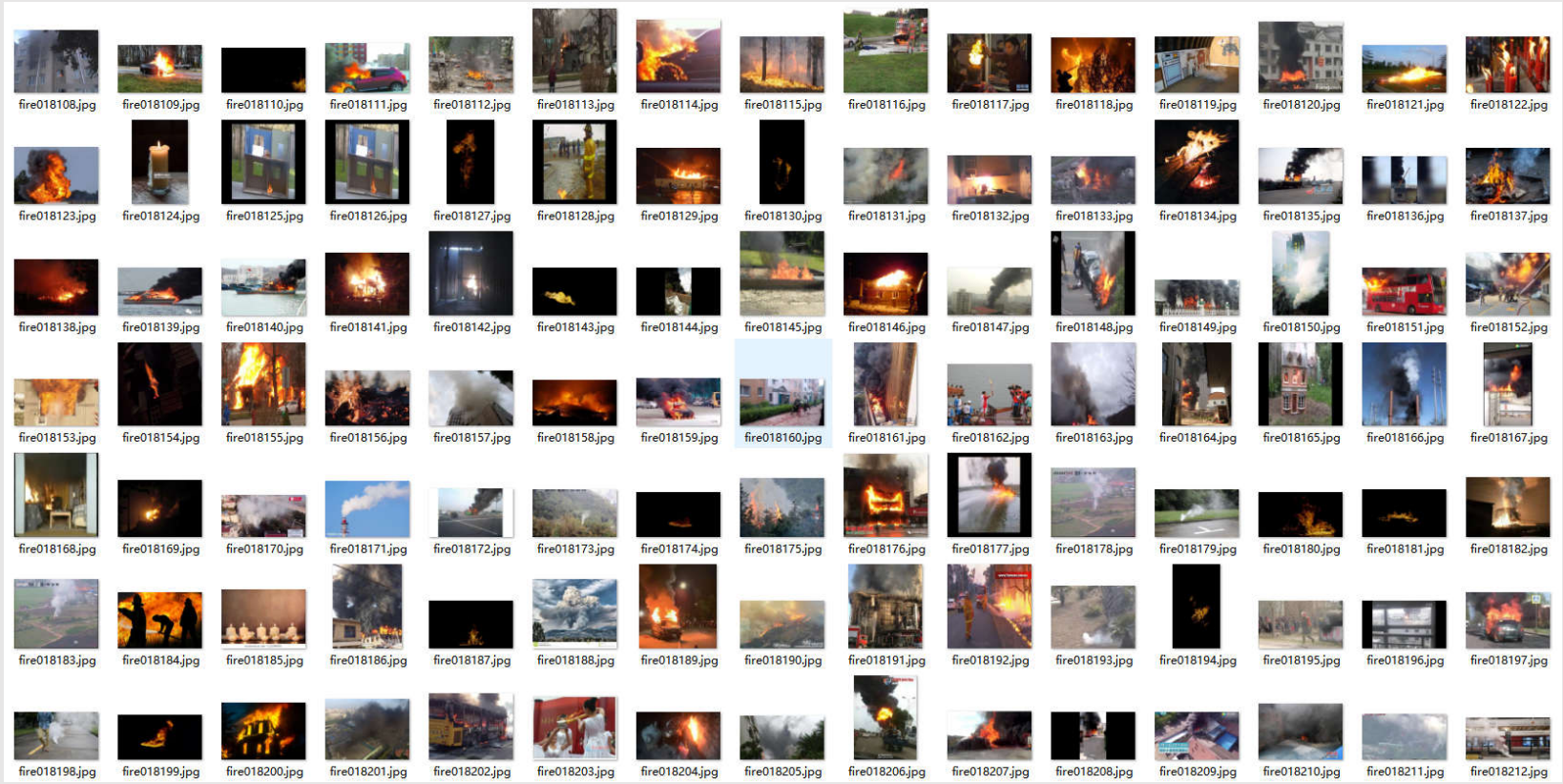
针对火焰视频的检测结果，将每帧得到的结果统计，组成火势发展初始序列，通过数据的预处理后利用序列分段与线性拟合算法，获得火势发展的重要转折点与主要的趋势变化。

第二部分

火情检测网络

- 数据集建立
- 网络模型选择
- 网络改进
- 实验验证

2.1 数据集建立 | 图像搜集



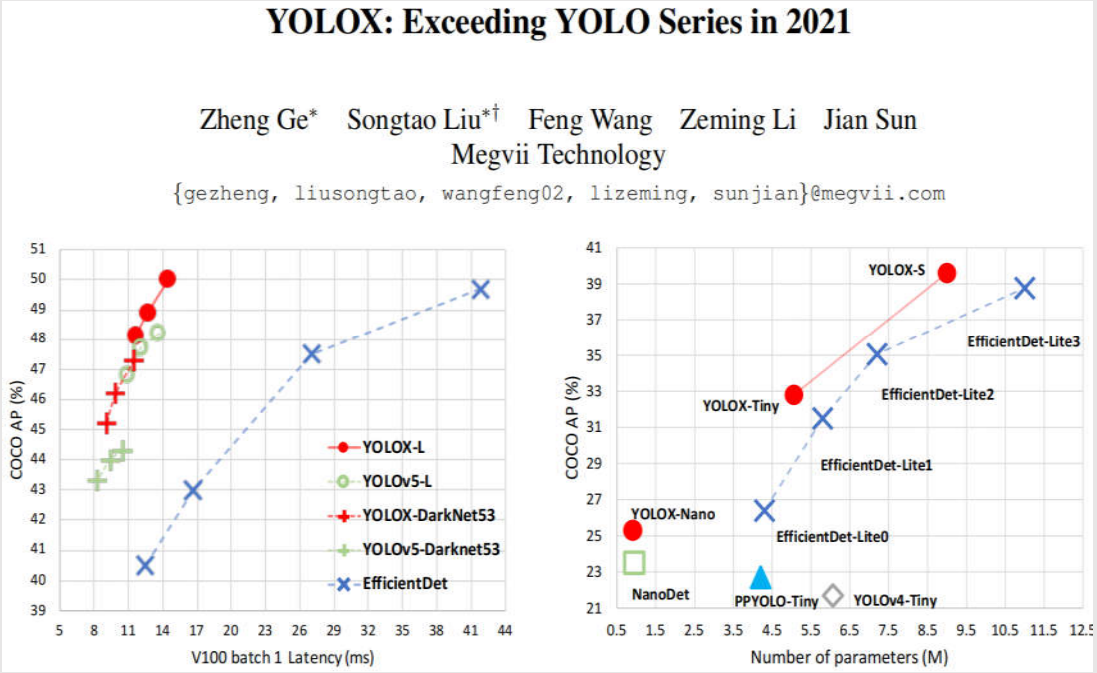
数据集的好坏严重影响着网络学习的最终效果。为了满足数据集的普适性，本论文中所建立的火焰数据集尽可能涵盖多个场景下的火情环境，包括室内、户外、丛林、山林、等烟火图像集、视频集，**共收集到19819张图像**。

2.1 数据集建立 | 数据标注

对于搜集到的数据集需要进一步完成标注工作，以方便用于深度学习网络训练。借助图片标注工具 LabelImg，按照PASCAL VOC数据集的格式要求，对数据集中的每一张图像进行目标位置标注，火焰标签为fire，烟雾标签为smoke，生成标签文件 xml，最终的数据集中包含**火焰目标26530个，烟雾目标14404个**。

对火情数据集进行划分，其中随机选择500张图片组成测试集，剩余部分按照8：2的比例构成训练集和验证集，最终完成数据集准备工作。

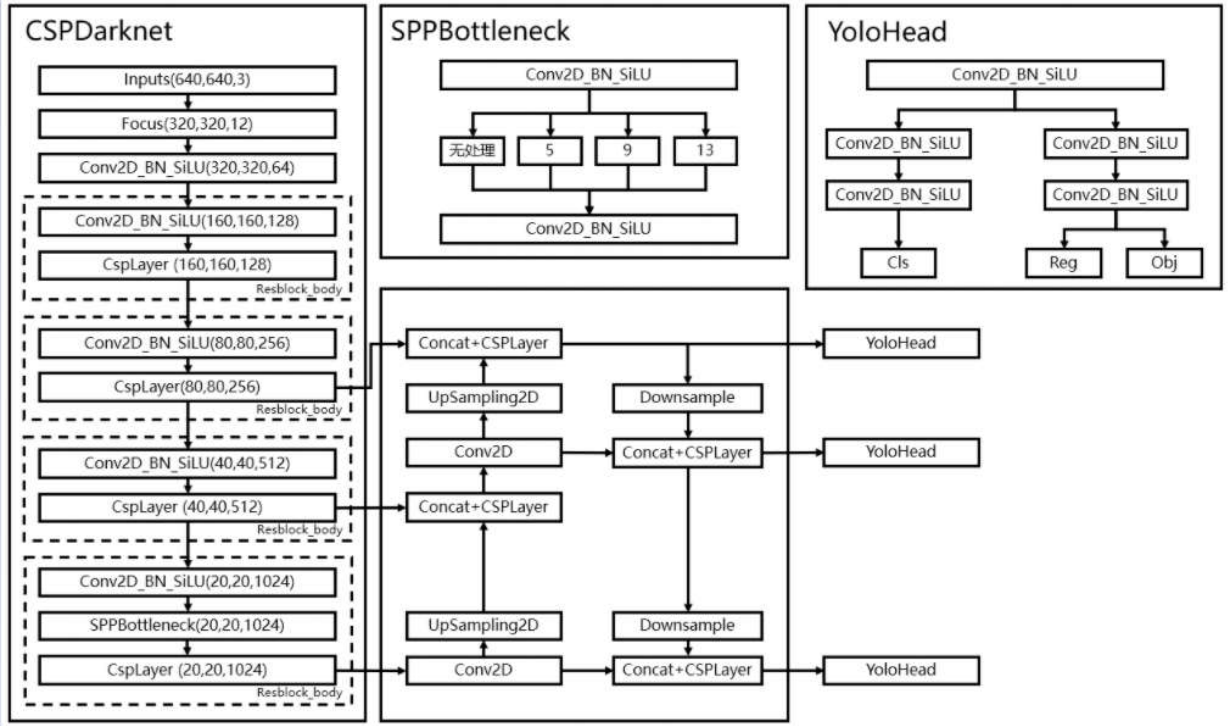
2.2 网络模型选择

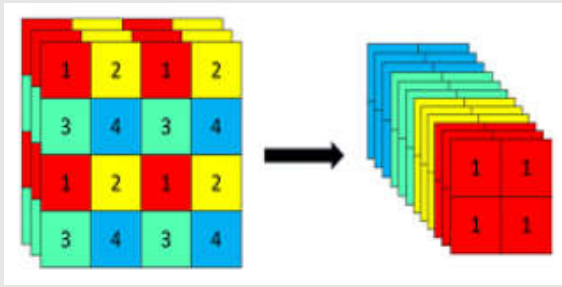


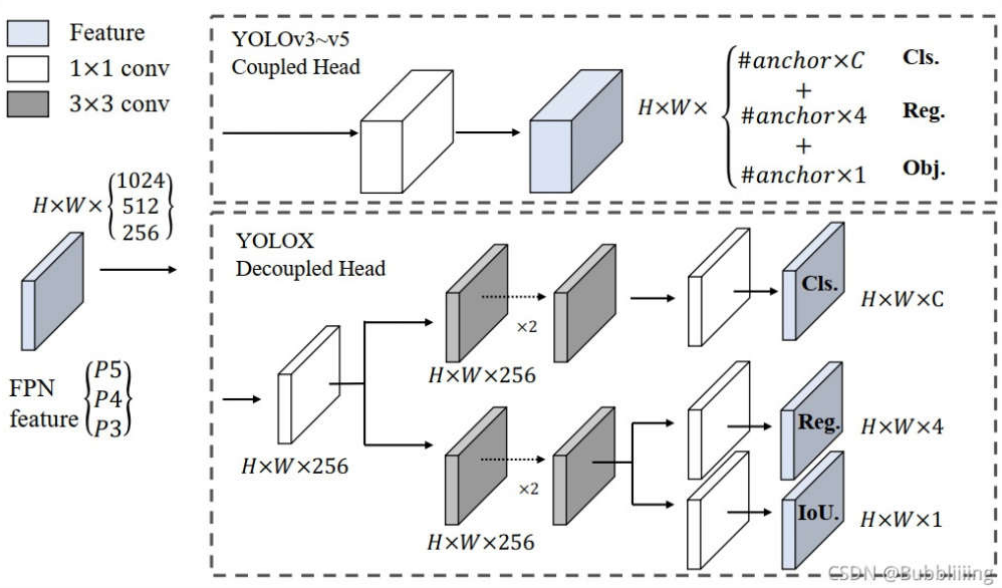
Method	Backbone	Size	FPS (V100)	AP (%)	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
YOLOv3 + ASFF* [18]	Darknet-53	608	45.5	42.4	63.0	47.4	25.5	45.7	52.3
YOLOv3 + ASFF* [18]	Darknet-53	800	29.4	43.9	64.1	49.2	27.0	46.6	53.4
EfficientDet-D0 [28]	Efficient-B0	512	98.0	33.8	52.2	35.8	12.0	38.3	51.2
EfficientDet-D1 [28]	Efficient-B1	640	74.1	39.6	58.6	42.3	17.9	44.3	56.0
EfficientDet-D2 [28]	Efficient-B2	768	56.5	43.0	62.3	46.2	22.5	47.0	58.4
EfficientDet-D3 [28]	Efficient-B3	896	34.5	45.8	65.0	49.3	26.6	49.4	59.8
PP-YOLOv2 [11]	ResNet50-vd-dcn	640	68.9	49.5	68.2	54.4	30.7	52.9	61.2
PP-YOLOv2 [11]	ResNet101-vd-dcn	640	50.3	50.3	69.0	55.3	31.6	53.9	62.4
YOLOv4 [1]	CSPDarknet-53	608	62.0	43.5	65.7	47.3	26.7	46.7	53.3
YOLOv4-CSP [30]	Modified CSP	640	73.0	47.5	66.2	51.7	28.2	51.2	59.8
YOLOv3-ultralytics[2]	Darknet-53	640	95.2	44.3	64.6	-	-	-	-
YOLOv5-M [7]	Modified CSP v5	640	90.1	44.5	63.1	-	-	-	-
YOLOv5-L [7]	Modified CSP v5	640	73.0	48.2	66.9	-	-	-	-
YOLOv5-X [7]	Modified CSP v5	640	62.5	50.4	68.8	-	-	-	-
YOLOX-DarkNet53	Darknet-53	640	90.1	47.4	67.3	52.1	27.5	51.5	60.9
YOLOX-M	Modified CSP v5	640	81.3	46.4	65.4	50.6	26.3	51.0	59.9
YOLOX-L	Modified CSP v5	640	69.0	50.0	68.5	54.5	29.8	54.5	64.4
YOLOX-X	Modified CSP v5	640	57.8	**51.2**	69.6	55.7	31.2	56.1	66.1

Models	AP (%)	Parameters	GFLOPs
YOLOv4-Tiny [30]	21.7	6.06 M	6.96
PPYOLO-Tiny	22.7	4.20 M	-
YOLOX-Tiny	**32.8** (+10.1)	5.06 M	6.45
NanoDet[3]	23.5	0.95 M	1.20
YOLOX-Nano	**25.3** (+1.8)	0.91 M	1.08

2.2 网络模型选择 | YOLOX







使用了SiLU激活函数，SiLU是Sigmoid和ReLU的改进版。SiLU具备无上界有下界、平滑、非单调的特性。SiLU在深层模型上的效果优于ReLU。可以看做是平滑的ReLU激活函数

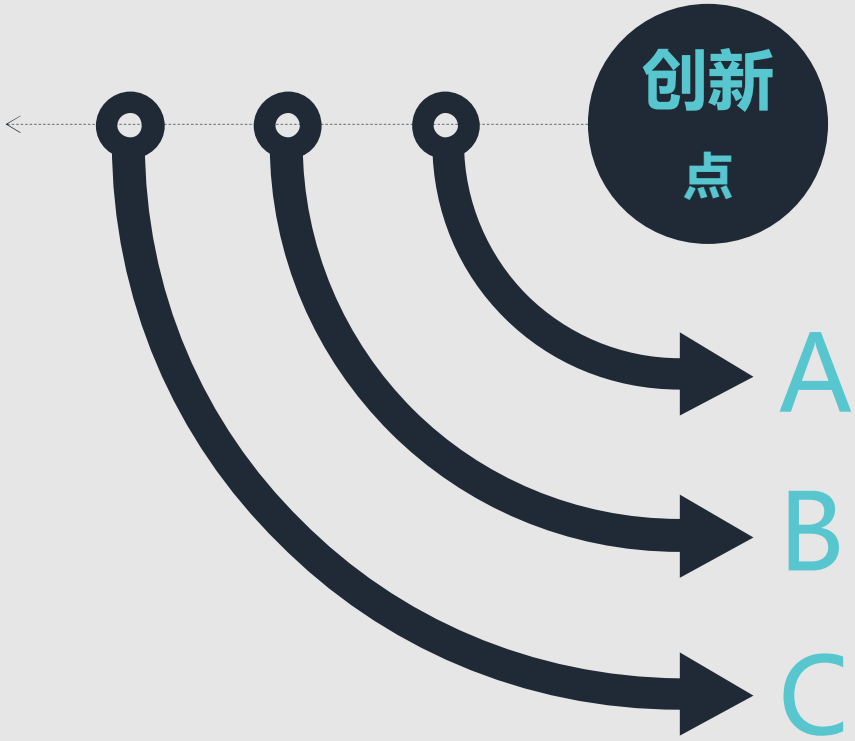
$$f(x) = x \cdot \text{sigmoid}(x)$$

CSDN @Bubbliiing

2.2 网络模型选择 | YOLOX不同型号的参数对比

时间	参数量/M	计算量/GFLOPs
Nano	0.9	1.1
Tiny	5.1	6.5
S	9.0	26.8
M	25.3	73.8
L	54.2	155.6
X	99.1	281.9

2.3 网络改进点



A 注意力机制+软阈值化

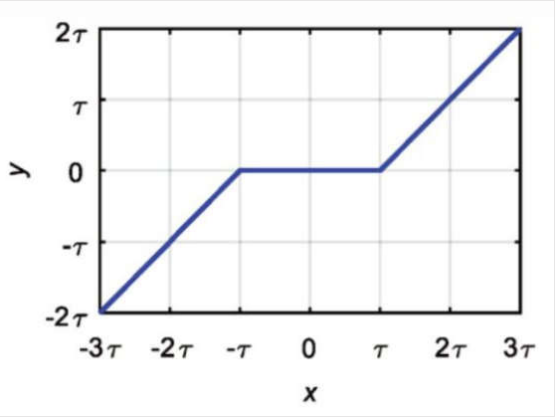
B FCS结构

C 可变形卷积

2.3.1 网络改进点 | 注意力机制+软阈值化

软阈值化

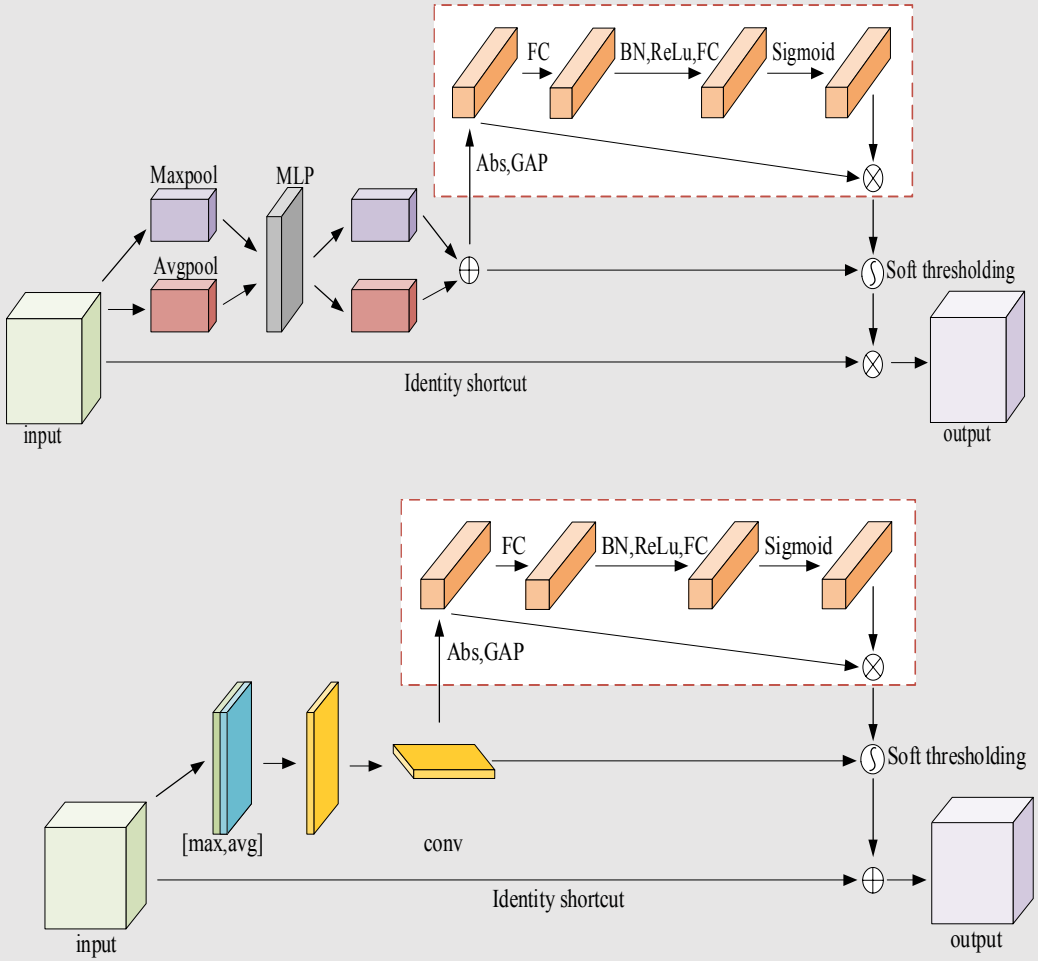
软阈值函数是大部分降噪方法的核心步骤。首先，我们需要设置一个正数阈值。该阈值不能太大，即不能大于输入数据绝对值的最大值，否则输出会全部为零。然后，软阈值函数会**将绝对值低于这个阈值的输入数据设置为零**。

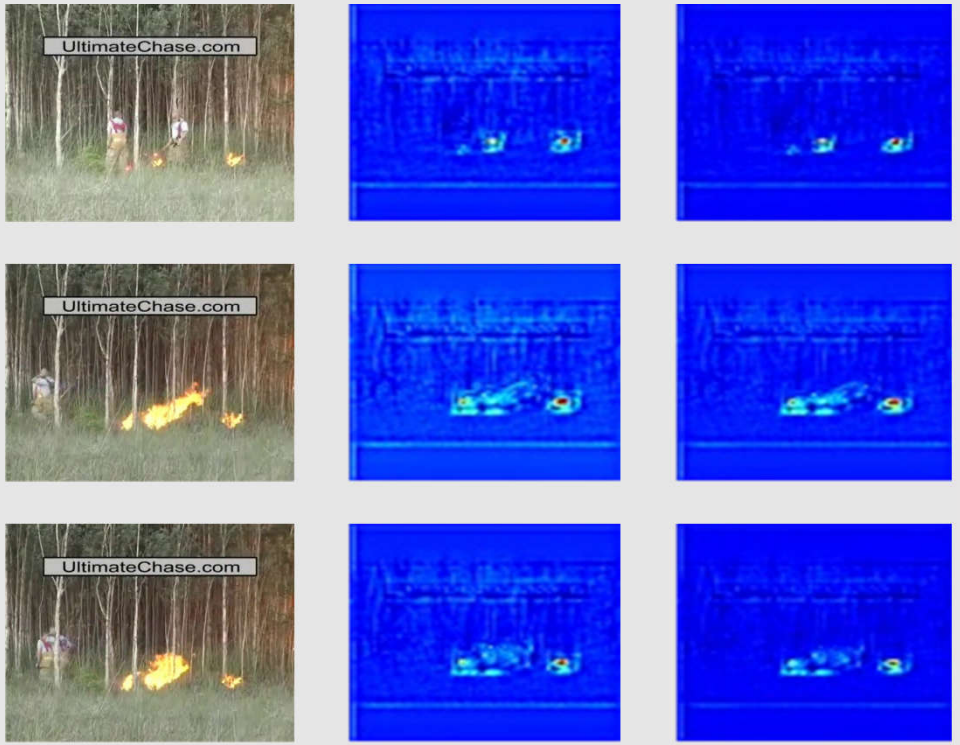


注意力机制

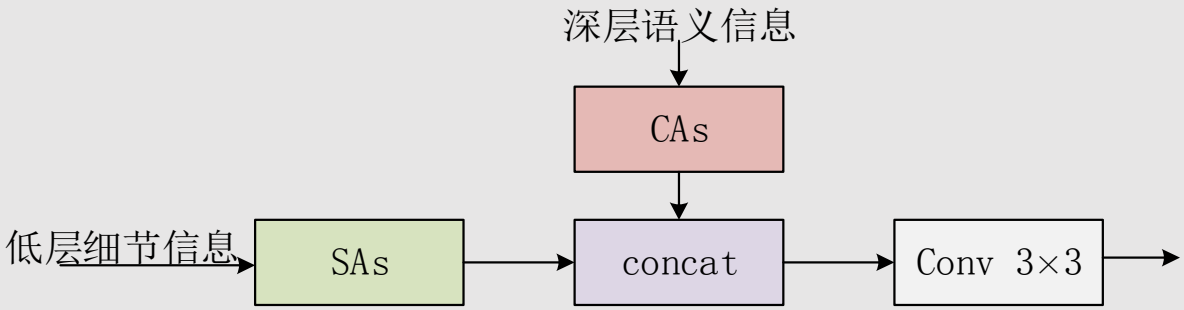
注意力机制是近年来深度学习领域的超级研究热点，其核心思想是评估各个特征通道的重要程度，然后根据其重要程度赋予各个特征通道合适的权重。首先，通过扫描全局信息，找到局部有用的信息；第二，**增强有用信息，抑制冗余信息**。另外，每个特征图的冗余信息内容往往是不同的，那么我们可以根据每个特征图的情况，通过注意力机制**自适应地设置不同的阈值**。

2.3.1 网络改进点 | 注意力机制+软阈值化



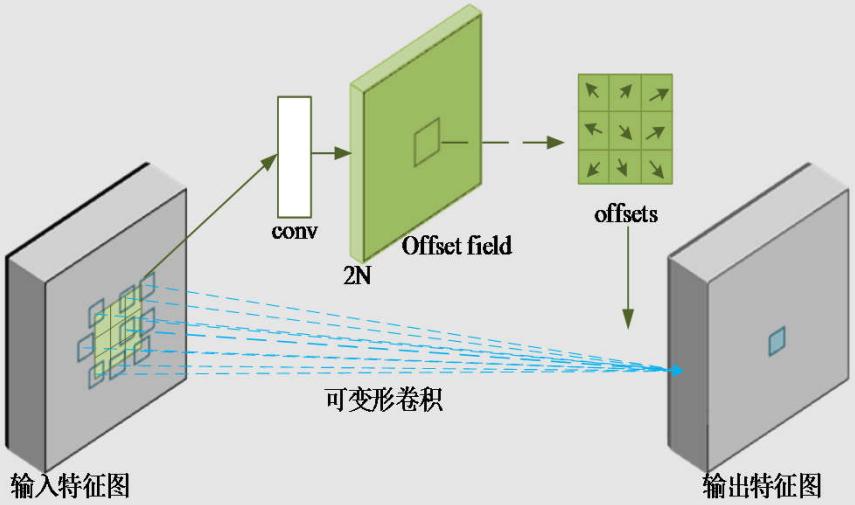


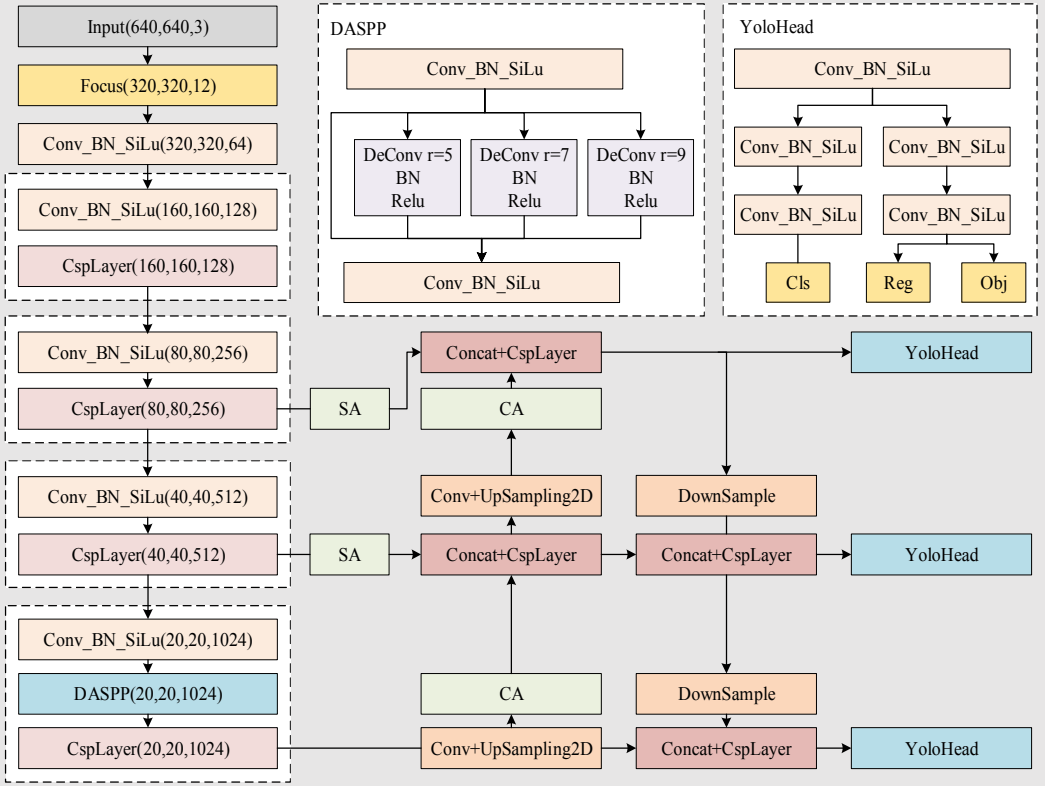
2.3.2 网络改进点 | FCS结构



由于**低层特征图包含丰富的边缘信息**，而**高层特征图具有更大的感受野**，得到的特征更抽象。关于注意力机制前面已经提到，CAs模块可以增强通道下特定语义的响应能力，SAs模块可以发现有较大贡献特征的空间信息。因此本文采用了FCS模块来融合高层和低层的特征，空间信息来自于特征提取网络之中，而通道信息来源于更深层的加强特征提取网络之中，来自于两个不同分支的特征进行concat融合操作，以提高检测准确性。

2.3.3 网络改进点 | 可变形卷积





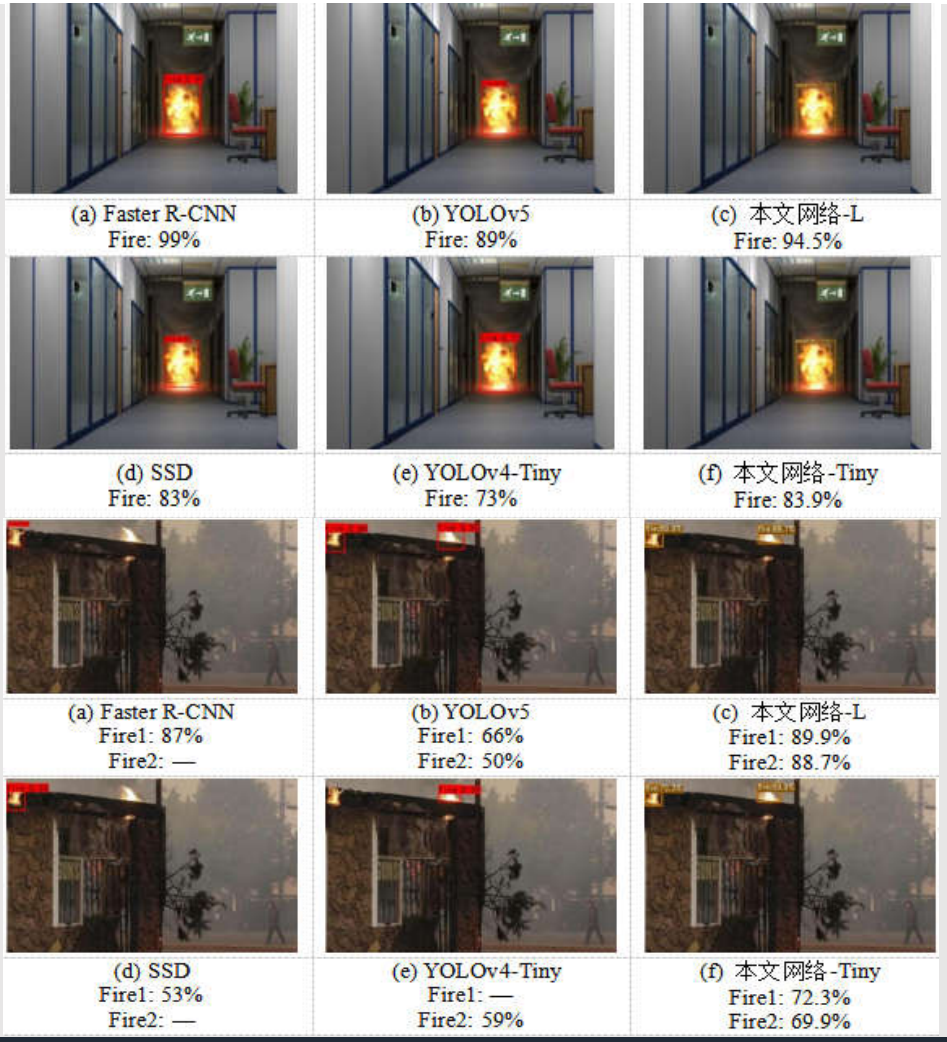
	原网络	改进网络
参数量	54,147,986	59498494
计算量/GFLOPs	77.68	79.53

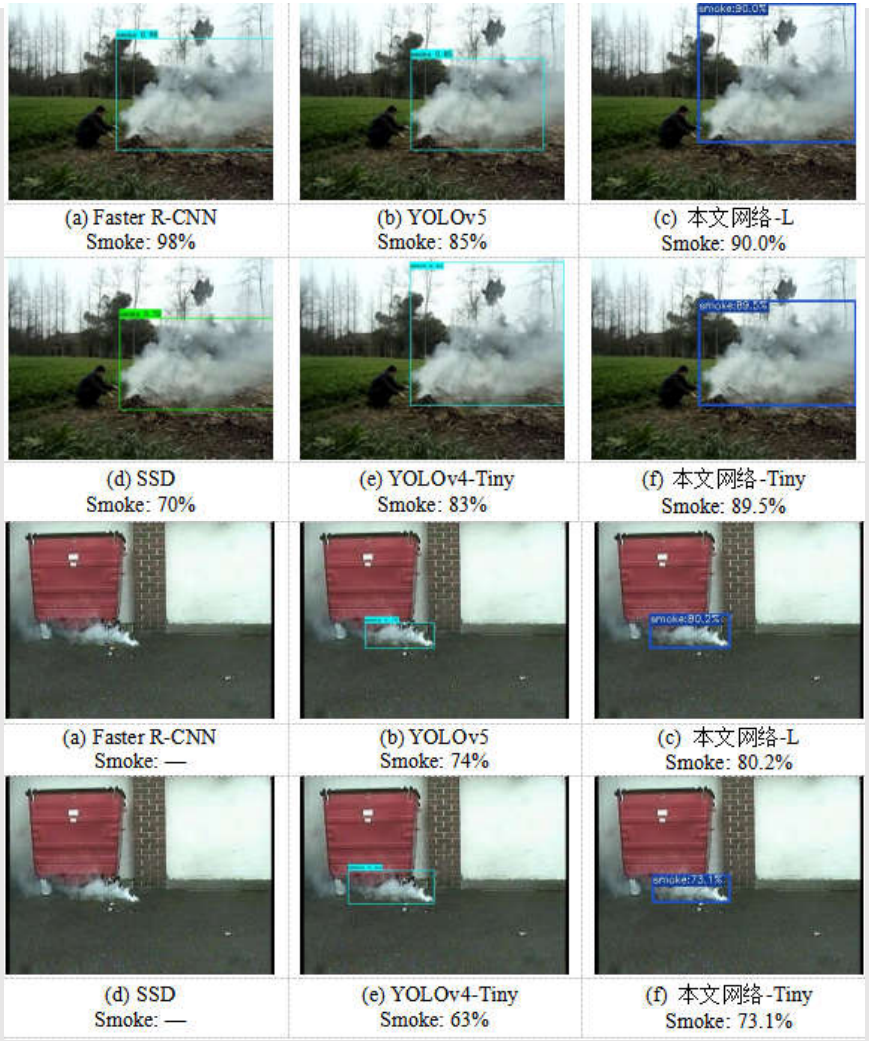
2.4 火情图片检测实验 | 数据验证

	Precision	Recall	F1	参数量	FPS
YOLOX	82.54%	59.36%	0.691	54,156,434	28.21
YOLOX-FCS	83.35%	61.08%	0.705	54,872,502	28.51
本文网络-L	85.43%	63.72%	**0.730**	59,498,494	25.04

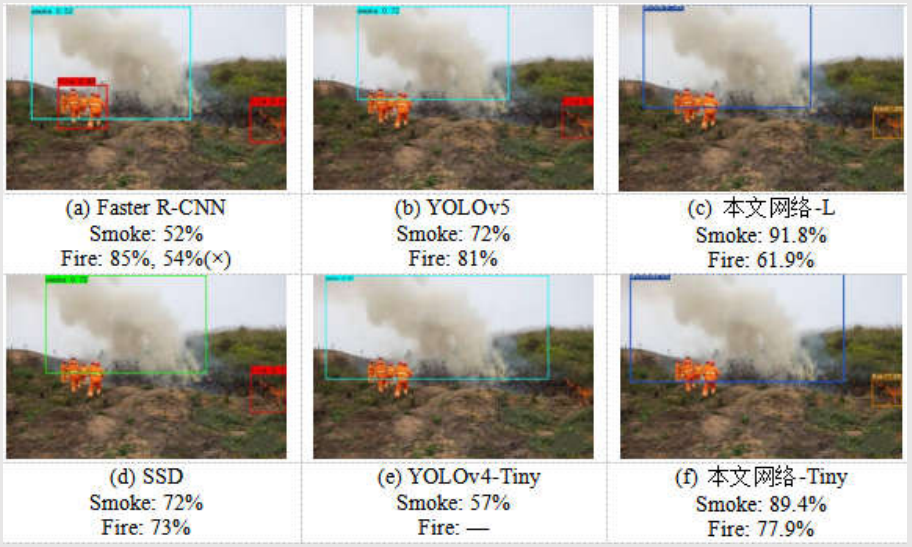
网络类型	火情检测网络	Precision	Recall	F1	参数量	FPS
基于候选区域类	Faster-RCNN	88.22%	65.29%	0.751	137,078,239	22.33
基于回归类	YOLOv5	84.46%	60.80%	0.707	46,636,735	33.78
	SSD	80.69%	57.03%	0.668	26,151,824	105.43
轻量型	YOLOv4-Tiny	85.94%	42.01%	0.648	6,056,606	129.04
	本文网络-Tiny	85.83%	55.80%	0.676	5,151,782	111.12

2.4 火情图片检测实验 | 直观验证





2.4 火情图片检测实验 | 直观验证



(a) Faster R-CNN
Smoke: 52%
Fire: 85%, 54%(×)

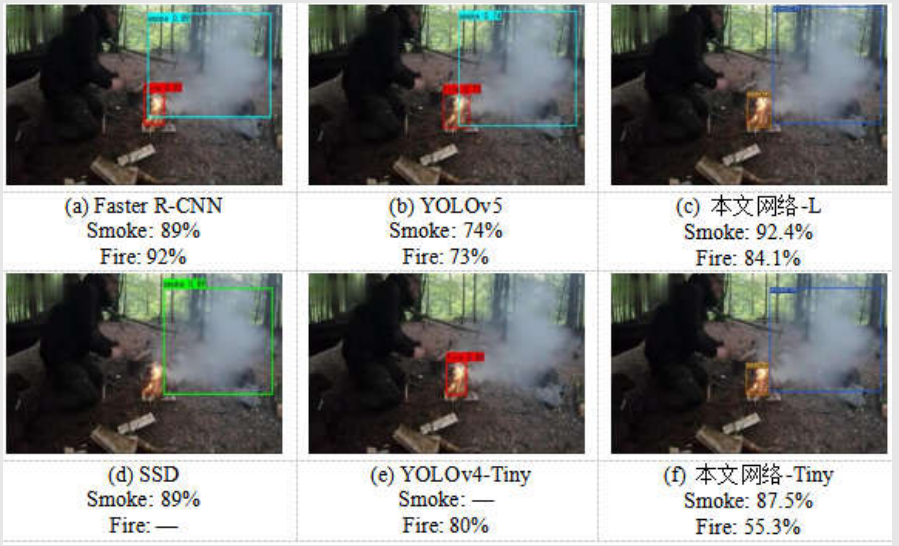
(b) YOLOv5
Smoke: 72%
Fire: 81%

(c) 本文网络-L
Smoke: 91.8%
Fire: 61.9%

(d) SSD
Smoke: 72%
Fire: 73%

(e) YOLOv4-Tiny
Smoke: 57%
Fire: —

(f) 本文网络-Tiny
Smoke: 89.4%
Fire: 77.9%



(a) Faster R-CNN
Smoke: 89%
Fire: 92%

(b) YOLOv5
Smoke: 74%
Fire: 73%

(c) 本文网络-L
Smoke: 92.4%
Fire: 84.1%

(d) SSD
Smoke: 89%
Fire: —

(e) YOLOv4-Tiny
Smoke: —
Fire: 80%

(f) 本文网络-Tiny
Smoke: 87.5%
Fire: 55.3%

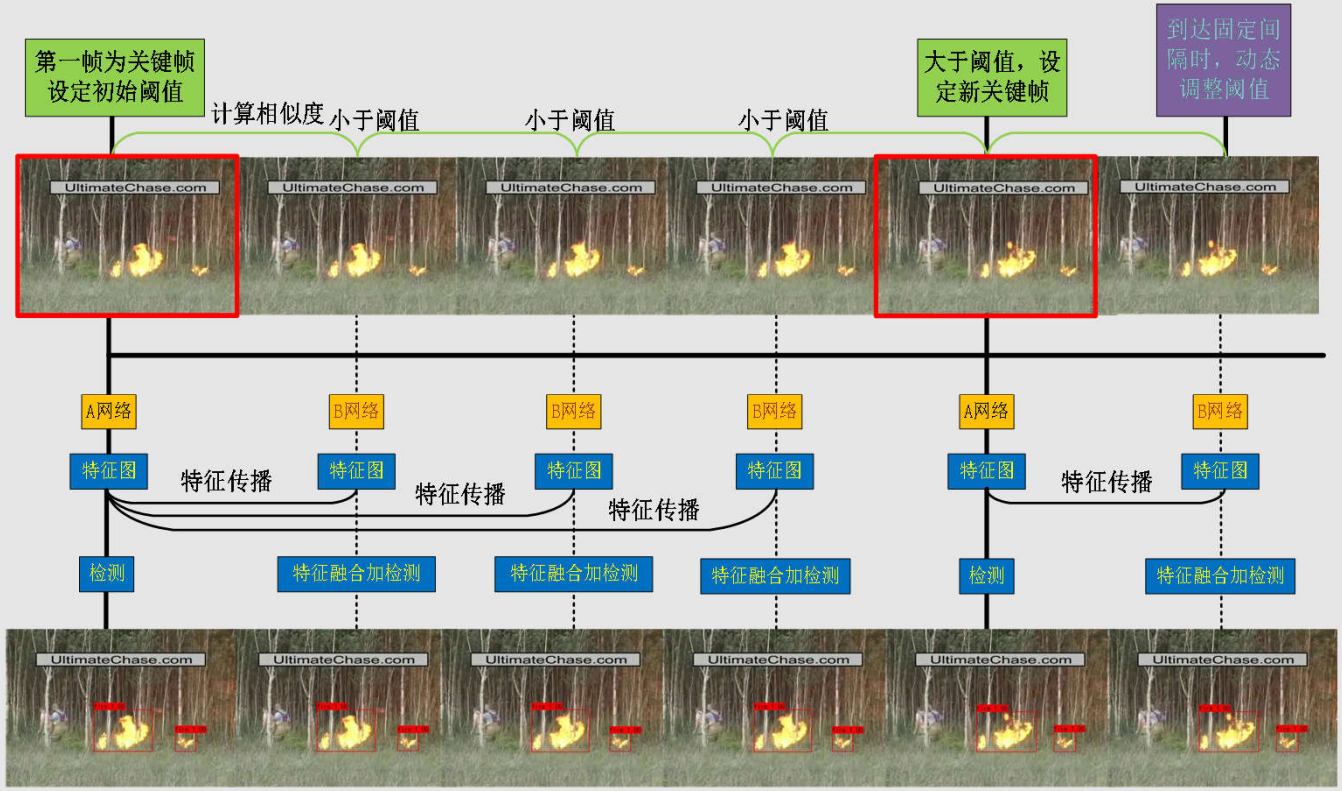
第三部分

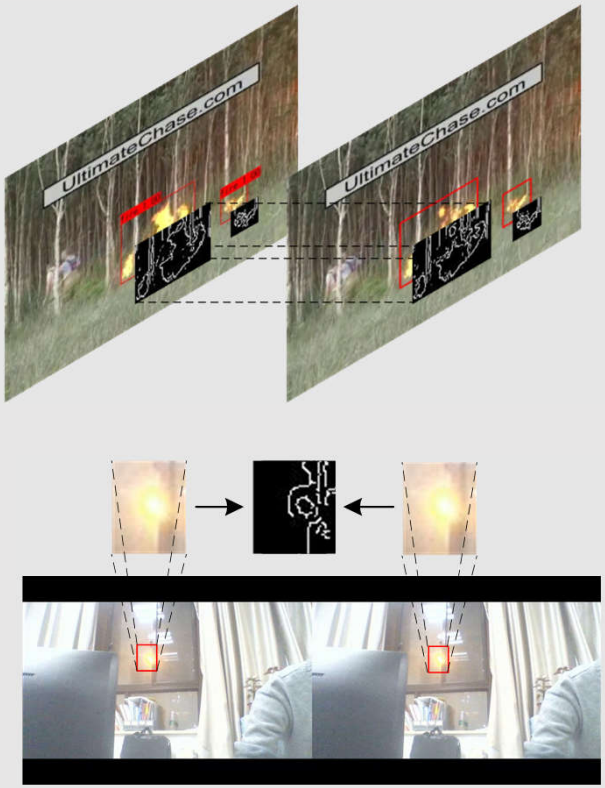
加速检测算法

- 视频图像特点
- 双网络特征融合算法
- 特征融合
- 实验验证

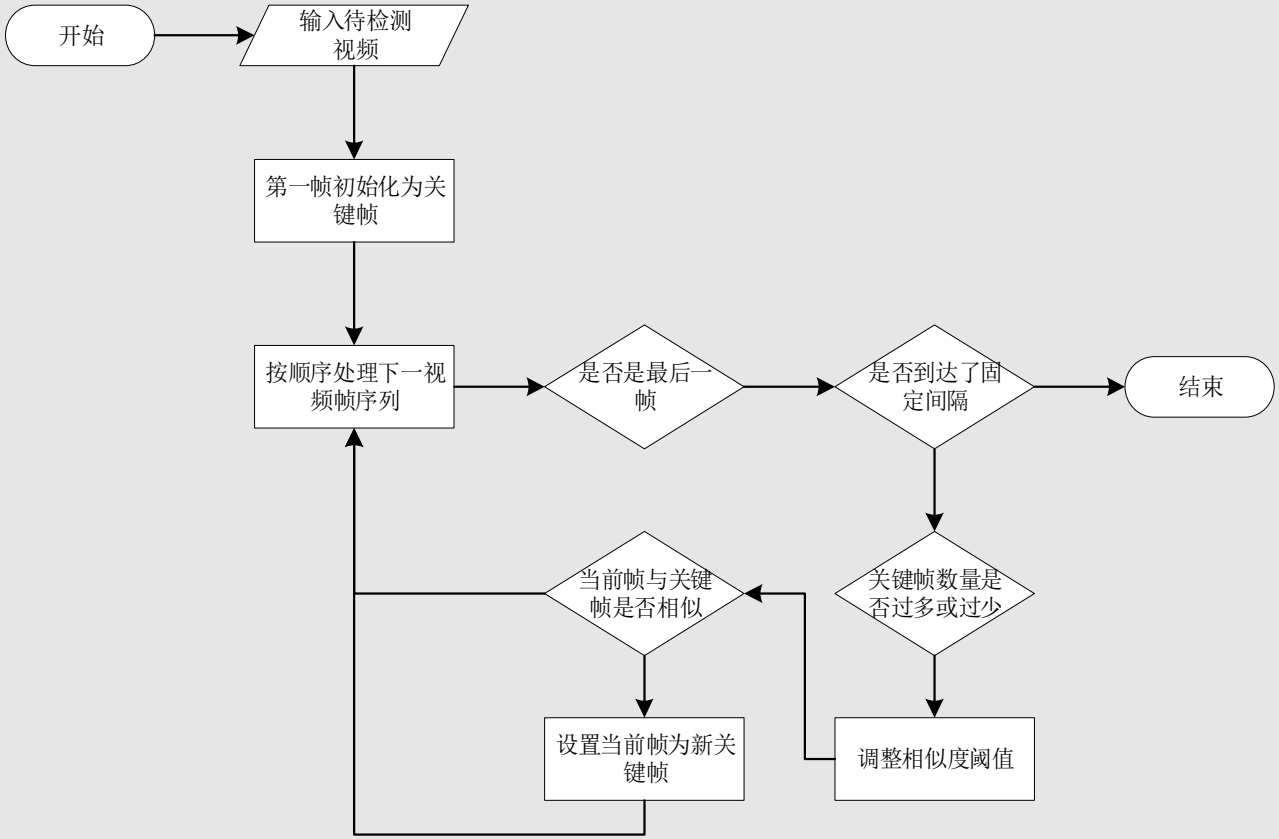
3.1 视频图像特点

3.2 双网络特征融合算法 | TFF算法





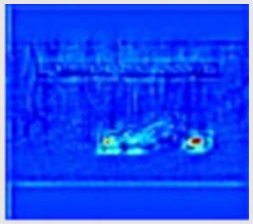
3.2 双网络特征融合算法 | 算法流程图



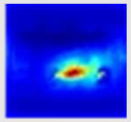
3.2 双网络特征融合算法 | 关键帧选取结果

3.3 特征融合

小目标



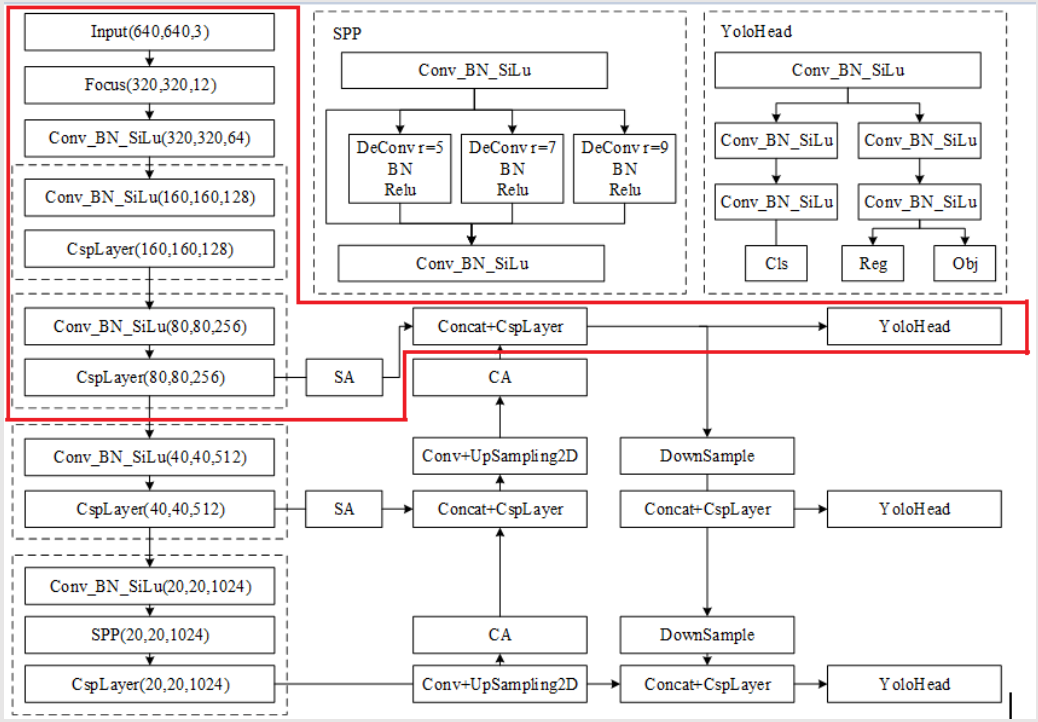
中目标

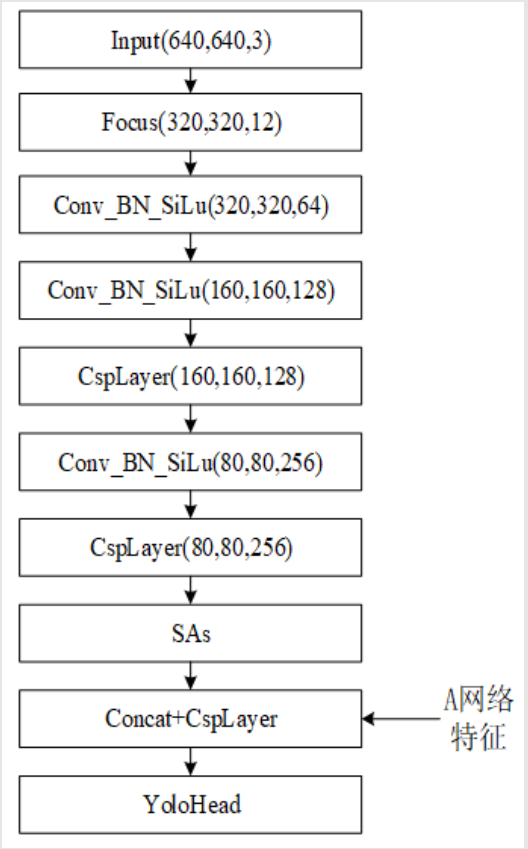


大目标

一般**浅层特征本就包括丰富的边界细节信息**，其正好也适合用于检测时间间隔很短的邻近帧之间**火焰的微小形变**，关于其欠缺的高层语义信息可以由A网络来补充。

3.3 特征融合 | B网络结构





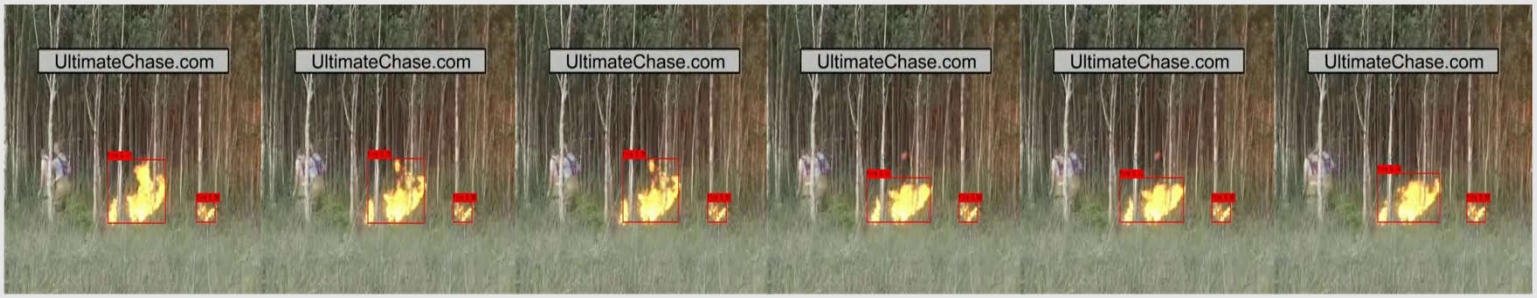
本文网络为例，本文网络-L的主干特征提取部分总参数量为27,075,968，预测部分为27,072,018，
而浅层网络的特征提取部分参数量仅为2,144,640，因此冗余帧的检测参数量**被缩减至原来的53.96%**

3.4 实验验证

3.4 实验验证

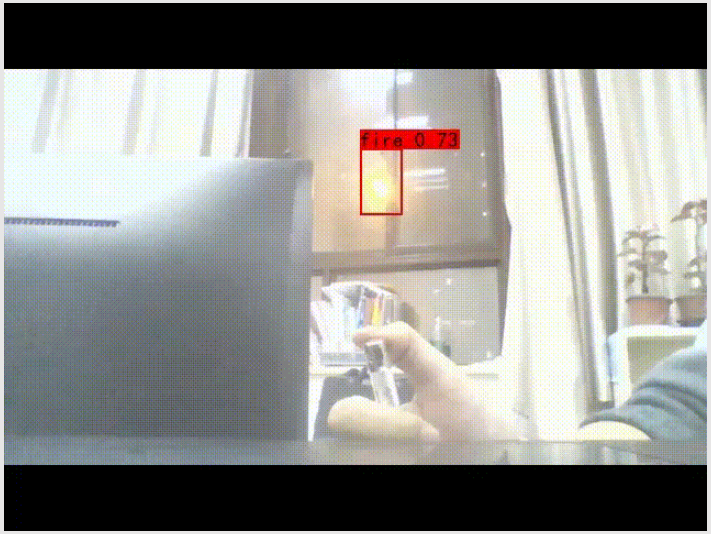
	Precision	Recall	F1	关键帧数量	检测总时长/s	FPS
本文网络	93.6%	87.35%	0.90	—	15.9	16.3
TFF算法	91.35%	84.48%	0.87	42	11.1	23.4

本文网络



TFF算法

3.4 实验验证 | 误检测试



3.4 实验验证 | 速度测试

测试视频	TFF算法				本文网络	
	关键帧数量	关键帧占比	检测总时长	mFPS	检测总时长	mFPS
VC1	72	28.2%	12.07	21.13	17.46	14.61
VC2	42	17.5%	10.58	22.68	16.66	14.41
VC3	629	52.4%	53.16	22.57	77.11	15.56
VC4	90	45.0%	7.75	25.81	10.71	18.67
VC5	372	95.4%	20.32	19.19	25.89	15.06
VC6	897	89.7%	45.68	21.89	51.89	19.27
VC7	952	95.2%	46.22	21.64	51.43	19.44
VC8	54	14.1%	18.93	20.29	28.13	13.65
平均值	—	—	—	21.90	—	16.33

第四部分

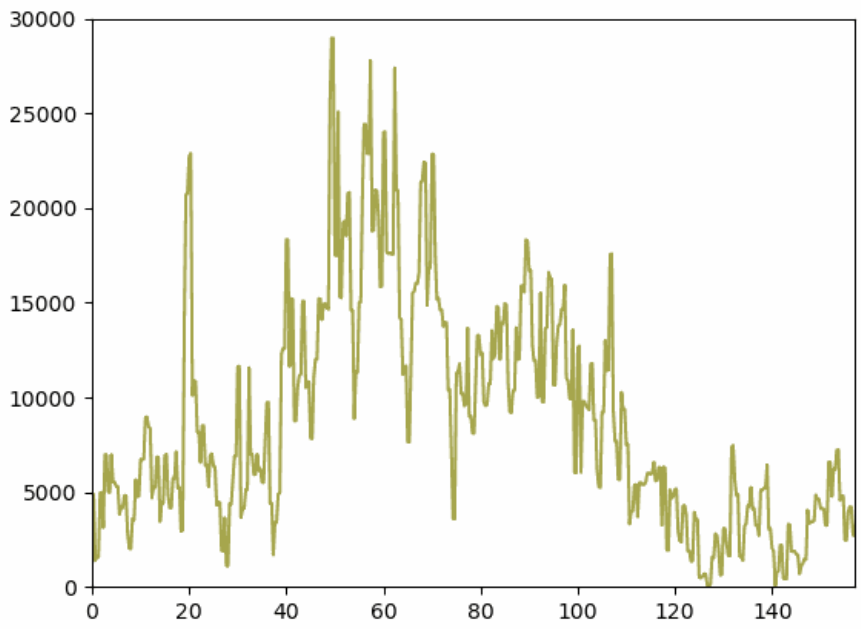
火情分析

- 问题分析
- 数据平滑
- 异常值处理
- 分段处理
- 实验验证

4.1 问题分析

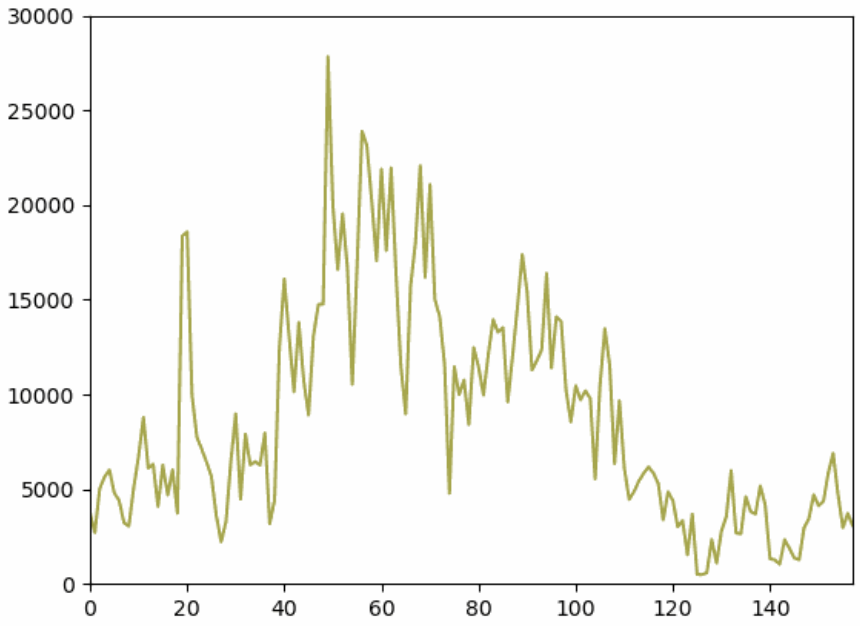
分析问题	分析指标	分段方法	数据预处理
一般的，当火焰增长没有受到太多外部环境干扰时，会经历“初起—充分发展—衰减”三个主要的阶段。对于火情的分析，应当指出火势处于什么阶段，或是在哪些时间段内经历了什么阶段，火势何时达到了最大？其整体的趋势又是如何？	如果将视频检测结果组成火势曲线，其波动正好对应了火势的变化，该曲线上的斜率正好可以代表火焰的变化。因此如何获取主要的趋势成为关键，如果可以将序列进行分段处理对各个分段进行拟合，即能表达火势的发展。	分段必须根据趋势的主要变化点，转折点分开。考虑从曲线上波动变化最大的点进行分析。 分段的数量也是关键。	能否直接将火焰检测结果后的数据进行分段？若是数据量过大，曲线波动太多是否会影响分段算法，另外如果某一帧检测结果有误，其值过大或过小，成为了一个虚假的分段点？

4.2 数据平滑



对火势发展情况进行分析，首先要对火焰视频分割结果中每帧得到的预测框面积进行统计，将其组成火势发展初始序列，如左图所示。初始序列一般较为粗糙，并且我们的关注点主要是火势的变化趋势，而不是每帧的局部信息，因此本文采用每隔3帧取火焰像素平均值的操作对初始序列进行压缩平滑处理，以减轻序列的抖动情况。压缩平滑后的序列毛刺较少，并保留了原有的火势发展趋势。

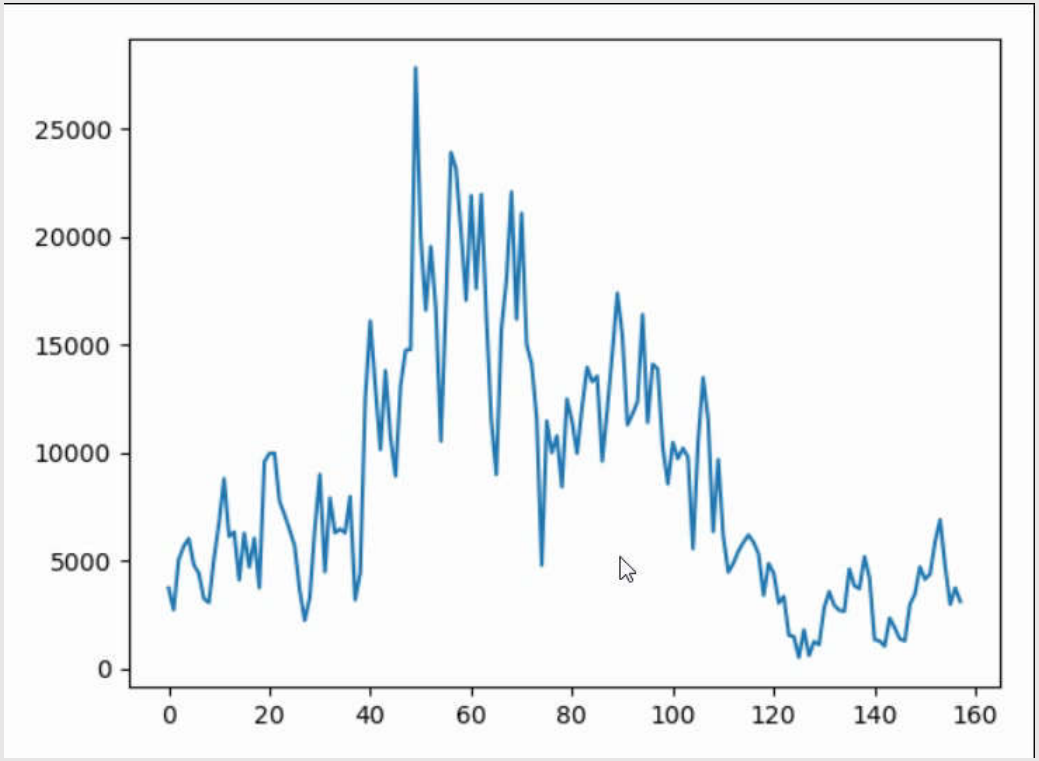
4.3 异常值处理



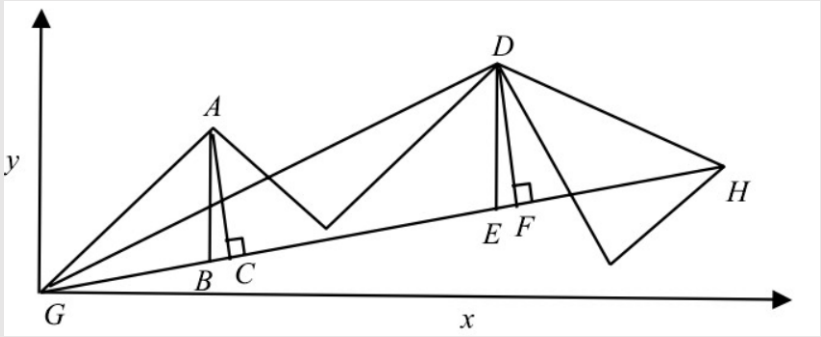
由于检测过程中偶尔会出现误检与漏检情况，进而导致该帧得到的结果明显区别于其临近帧的像素数。因此在不影响分析结果的前提下，可将某一帧突然增大或骤降的值定义为异常值。文中通过长度为5的窗口进行异常值处理，当某一帧的结果值大于该帧前后各两帧的平均值2倍或低于该平均值的0.1倍即视为异常值，将其重新赋为平均值。

$$\begin{cases} M = \sum_{\substack{k=i-2 \\ k\neq i}}^{i+2} f(k)/4 & i = 2,\ldots,n-2 \\ f(i) = M \quad if \;\; f(i) \geq 2\times M \;\; or \;\; f(i) \leq 0.1\times M \end{cases}$$

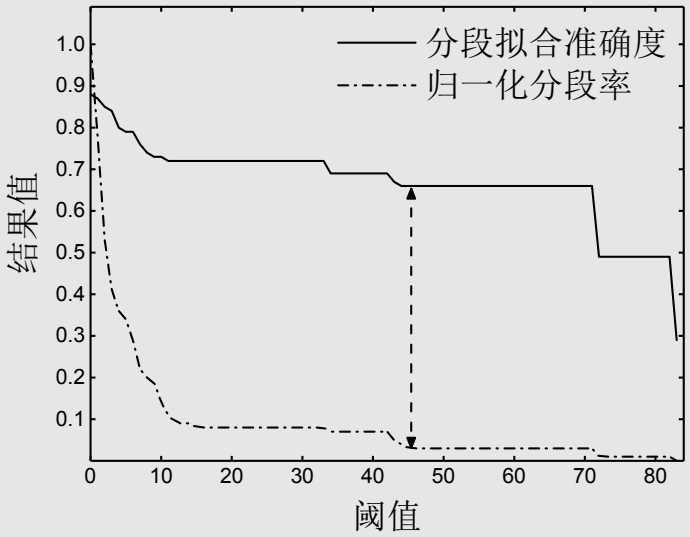
4.4 分段处理 | 寻找关键点



首先用直线连接当前序列的首尾两点，通过遍历当前序列，寻找与该直线最远垂直距离的点，若该距离不小于分段阈值时将其作为一个关键点Ii。然后将序列从Ii处划分为左子序列和右子序列，再利用相同的方法寻找左子序列中的关键点和右子序列的关键点，继续依次递归的划分左右子序列和寻找关键点，直到所有最远垂直距离小于初始设定的分段阈值时结束。



4.4 分段处理 | 分段阈值设定



$$Xs_i = \frac{(X_i - X_{min})}{(X_{max} - X_{min})}$$

$$A_i = 1 - \frac{\sum_{j=1}^{n} \left(\frac{|f(j) - y_j|}{y_j} \right)}{n}$$

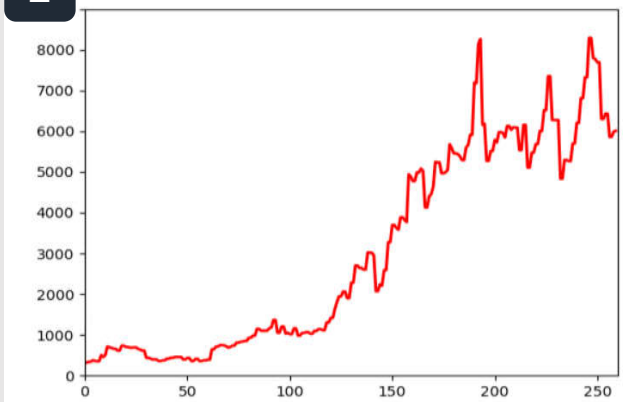
在上述序列分段算法中，分段阈值的设定决定了关键点的选取，也决定了分段序列拟合的结果，而分段阈值的设定需要人工估计，难以取得稳定的效果。分段阈值太小时，虽然对原序列的拟合准确度更高，但也会导致关键点和分段数量过多，无法提取出火情发展中的几个关键趋势；分段阈值太大时，得到的关键点又太少，无法准确地反映出火情发展趋势。因此，分段阈值的设定必须在分段数量与分段拟合准确度之间取得平衡。

由于分段数量与分段拟合准确度二者之间是正比关系，即随着分段阈值的不断增加，二者的值也会同步下降，但二者之间的差值是变化的。当某一阈值取到了最大差值时，则说明该阈值能够在高分段拟合准确度和低分段数量上取得最佳平衡，可以作为最终的分段阈值。

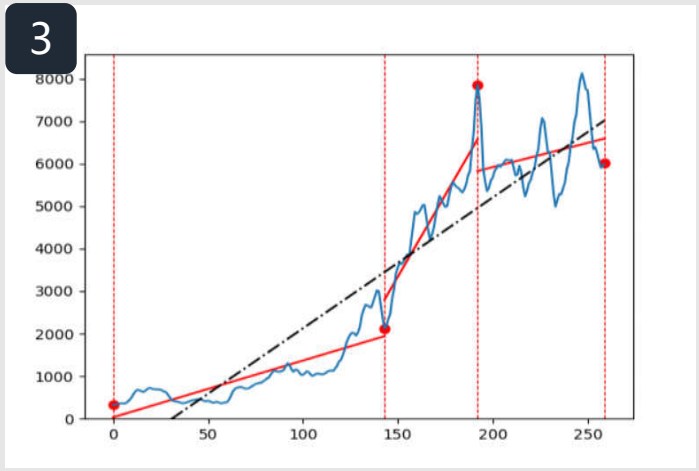
4.5 实验验证

1

2



3



4

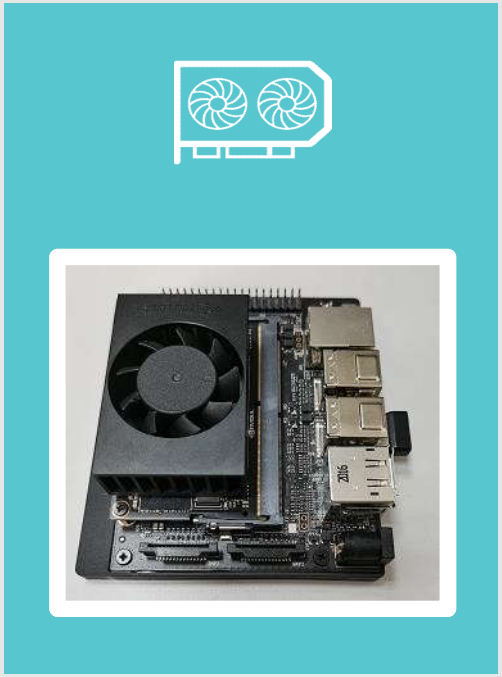
峰值	8125
均值	3032
最大时刻	16.1s
分段数量	3
分段拟合准确度/%	73.71
起火时刻	0s
熄火时刻	—
序列整体及各段拟合斜率	$K_0 = 30.79$ $K_1 = 13.35$ $K_2 = 77$ $K_3 = 11.43$

第五部分

平台搭建与测试

- 硬件环境
- 软件部分
- 系统展示

5.1 硬件环境

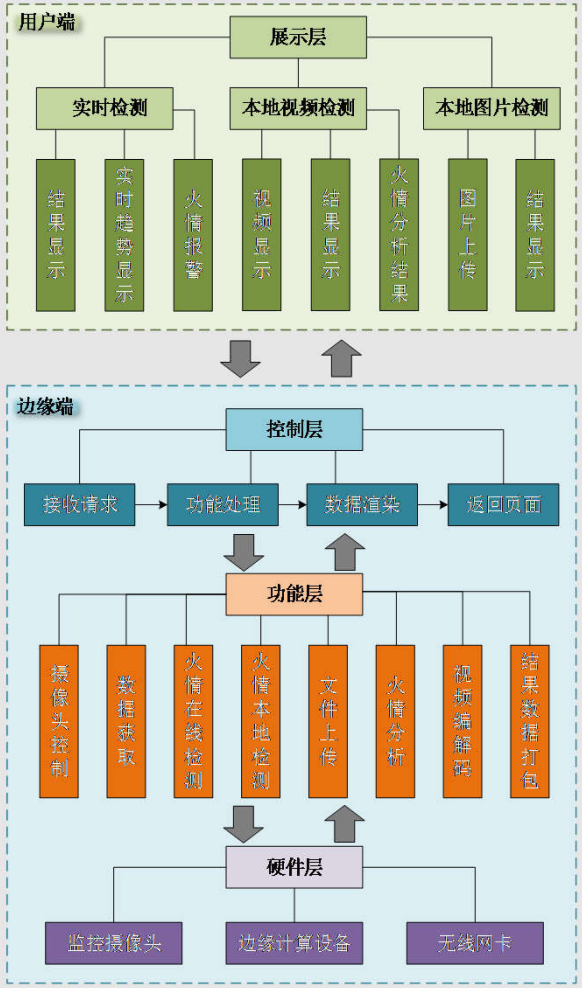


CPU	6-core NVIDIA Carmel ARM®v8.2 64-bit CPU 6 MB L2 + 4 MB L3
GPU	NVIDIA Volta architecture with 384 NVIDIA CUDA® cores and 48 Tensor cores
AI性能	6 TFLOPS(FP16) | 14-21 TOPS(INT8)
深度学习加速器	2x NVDLA Engines
视觉加速器	7-Way VLIW Vision Processor
Memory	8 GB 128-bit LPDDR4x @ 51.2GB/s
支持相机	2x MIPI CSI-2 DPHY lanes
网络连接	Gigabit Ethernet, M.2 Key E (WiFi/BT included), M.2 Key M (NVMe)
USB接口	4x USB 3.1, USB 2.0 Micro-B
规格尺寸	103 mm x 90.5 mm x 34.66 mm

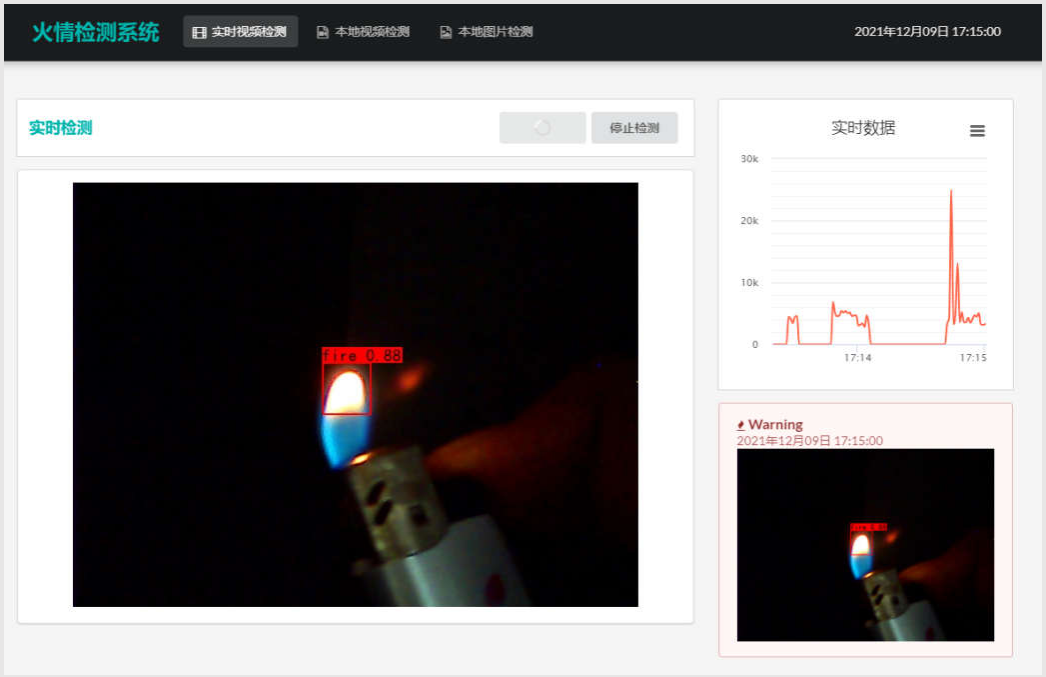
5.2 软件部分

	本文网络	mFPS	TFF算法检测	mFPS
VC1	43.28s	5.18	31.36	8.13
VC2	41.22	5.82	31.35	7.65
VC3	178.99	6.70	124.28	9.65
VC4	24.94	8.01	20.69	9.66
VC5	286.10	6.99	200.01	9.99
VC6	57.16	6.82	42.98	9.07
VC7	214.44	4.66	172.34	5.80
VC8	60.28	6.37	48.67	7.89
实时检测（1080P）	—	4.28	—	5.52
平均值	—	6.09	—	8.15

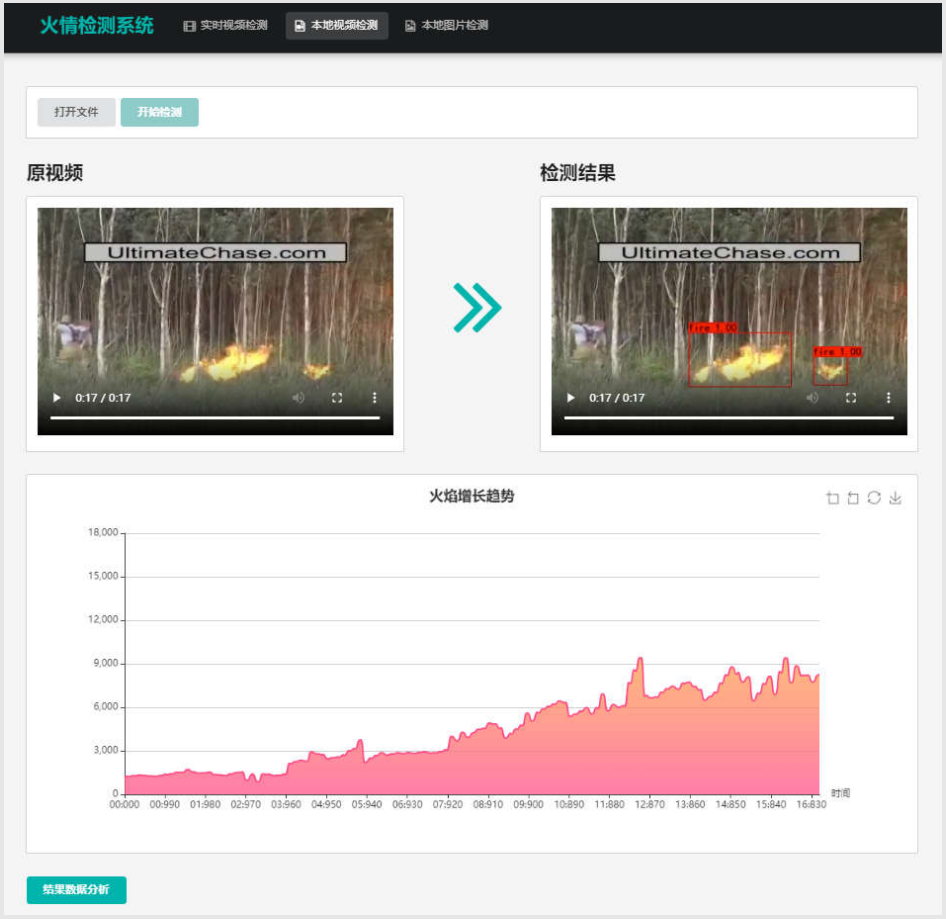
TFF算法比基于单帧检测的方法在边缘计算设备上提高了33.83%

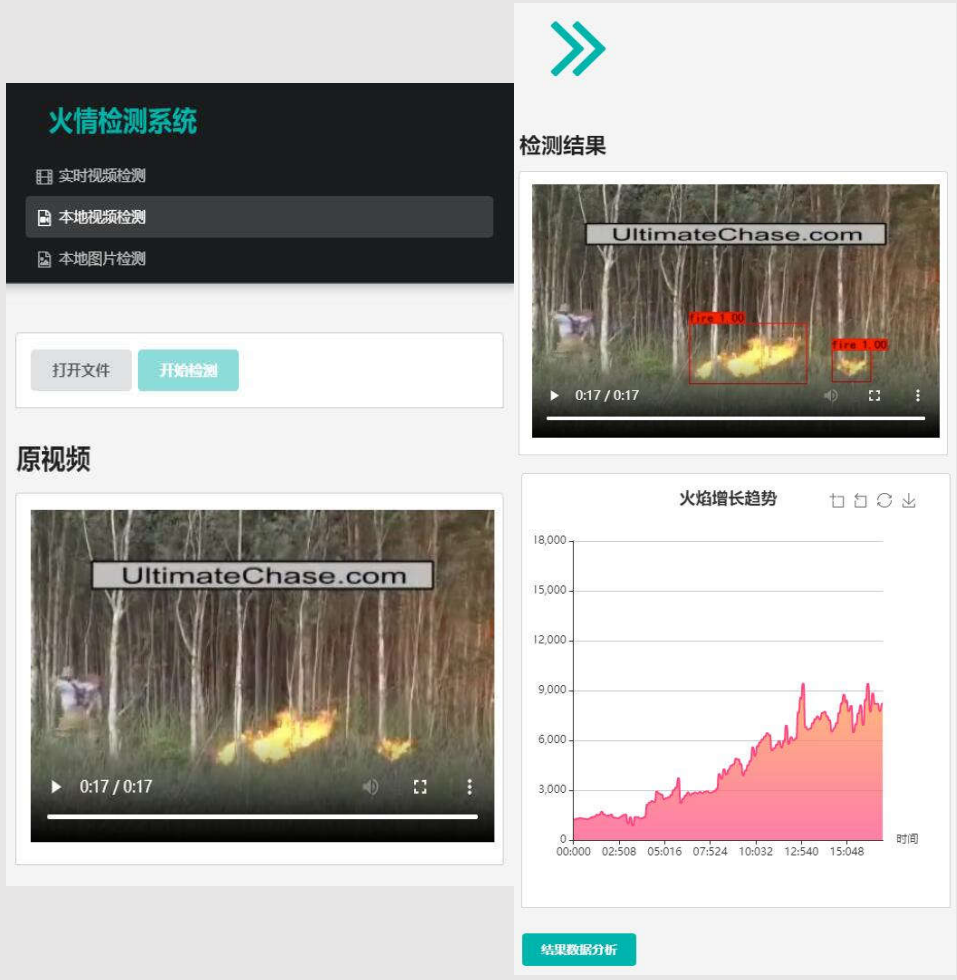


5.3 系统展示 | 实时视频检测

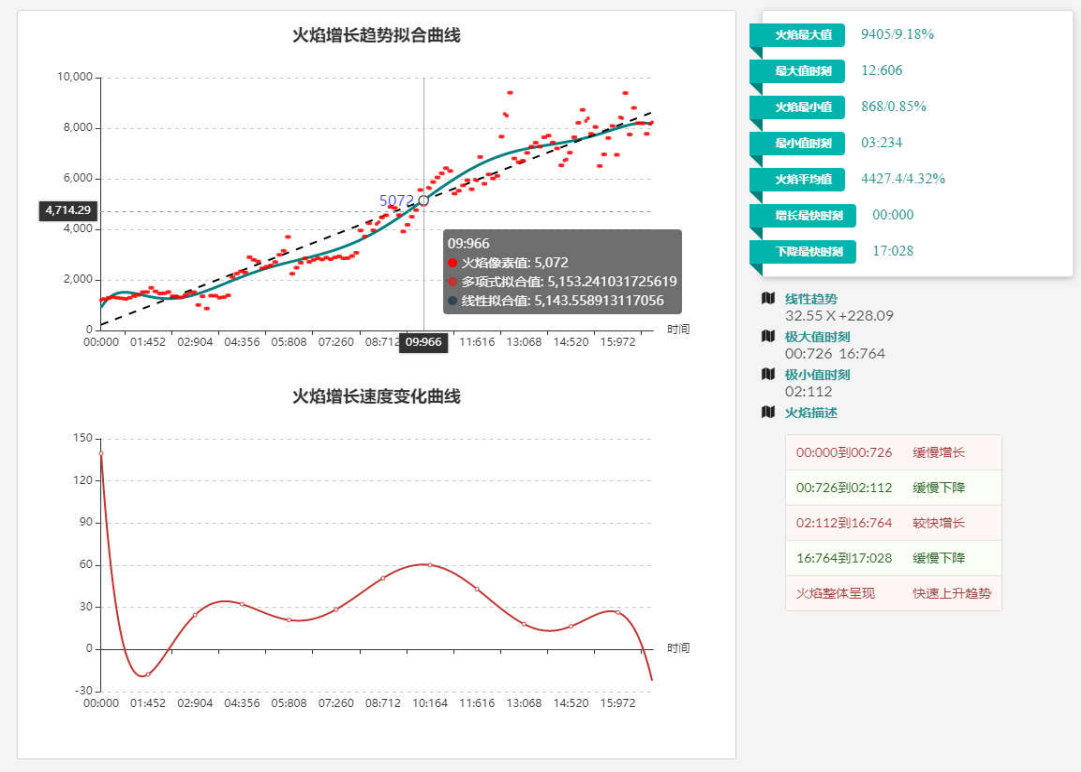


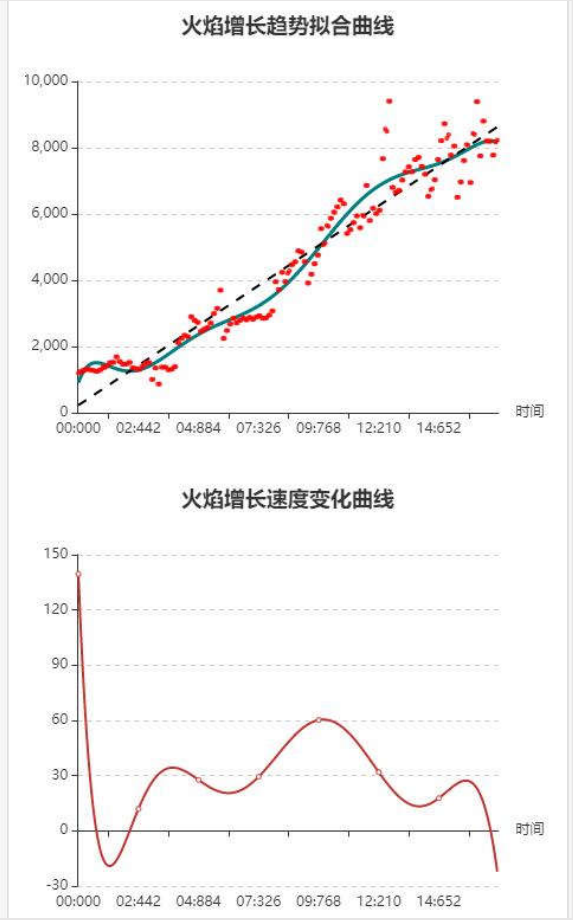
5.3 系统展示 | 本地视频检测



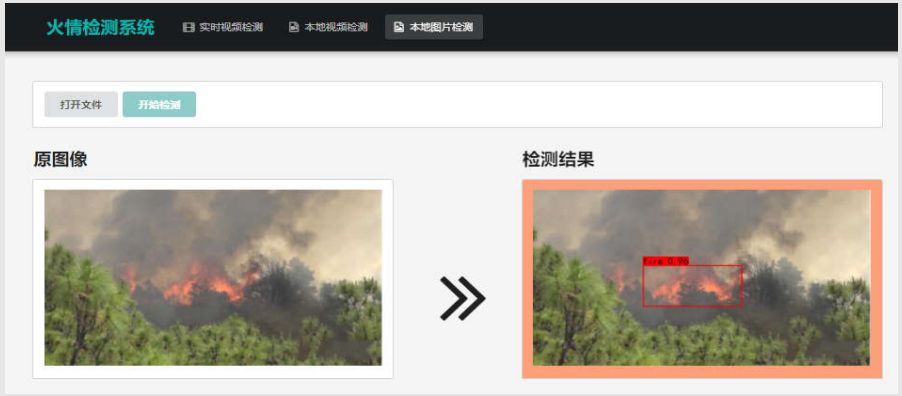


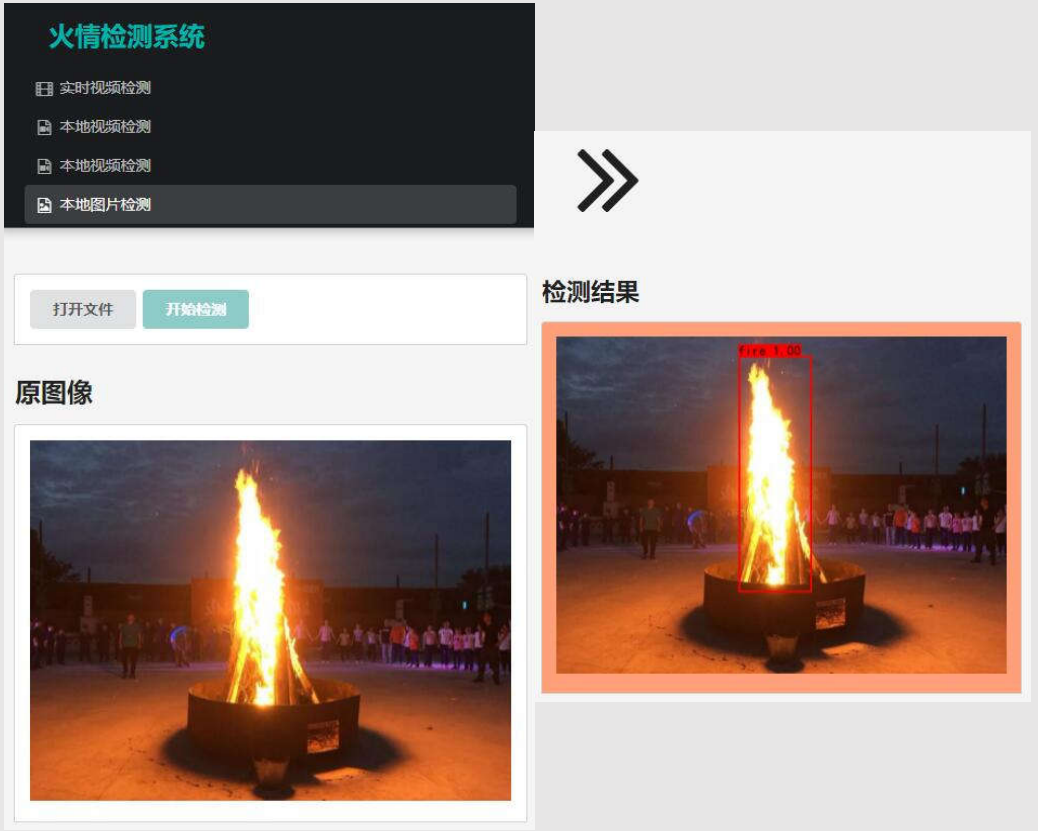
5.3 系统展示 | 火情分析


火焰增长趋势拟合曲线
10,000
8,000
6,000
4,000
2,000
0
4,714.29
5072
09:966
火焰像素值: 5,072
多项式拟合值: 5,153.241031725619
线性拟合值: 5,143.558913117056
00:000 01:452 02:904 04:356 05:808 07:260 08:712 09:966 11:616 13:068 14:520 15:972
时间
火焰增长速度变化曲线
150
120
90
60
30
0
-30
00:000 01:452 02:904 04:356 05:808 07:260 08:712 10:164 11:616 13:068 14:520 15:972
时间
火焰最大值 9405/9.18%
最大值时刻 12:606
火焰最小值 868/0.85%
最小值时刻 03:234
火焰平均值 4427.4/4.32%
增长最快时刻 00:000
下降最快时刻 17:028
线性趋势
32.55 X +228.09
极大值时刻
00:726 16:764
极小值时刻
02:112
火焰描述
00:000到00:726 缓慢增长
00:726到02:112 缓慢下降
02:112到16:764 较快增长
16:764到17:028 缓慢下降
火焰整体呈现 快速上升趋势


火焰增长趋势拟合曲线
10,000
8,000
6,000
4,000
2,000
0
00:000 02:442 04:884 07:326 09:768 12:210 14:652
时间
火焰增长速度变化曲线
150
120
90
60
30
0
-30
00:000 02:442 04:884 07:326 09:768 12:210 14:652
时间

5.3 系统展示 | 本地图片检测



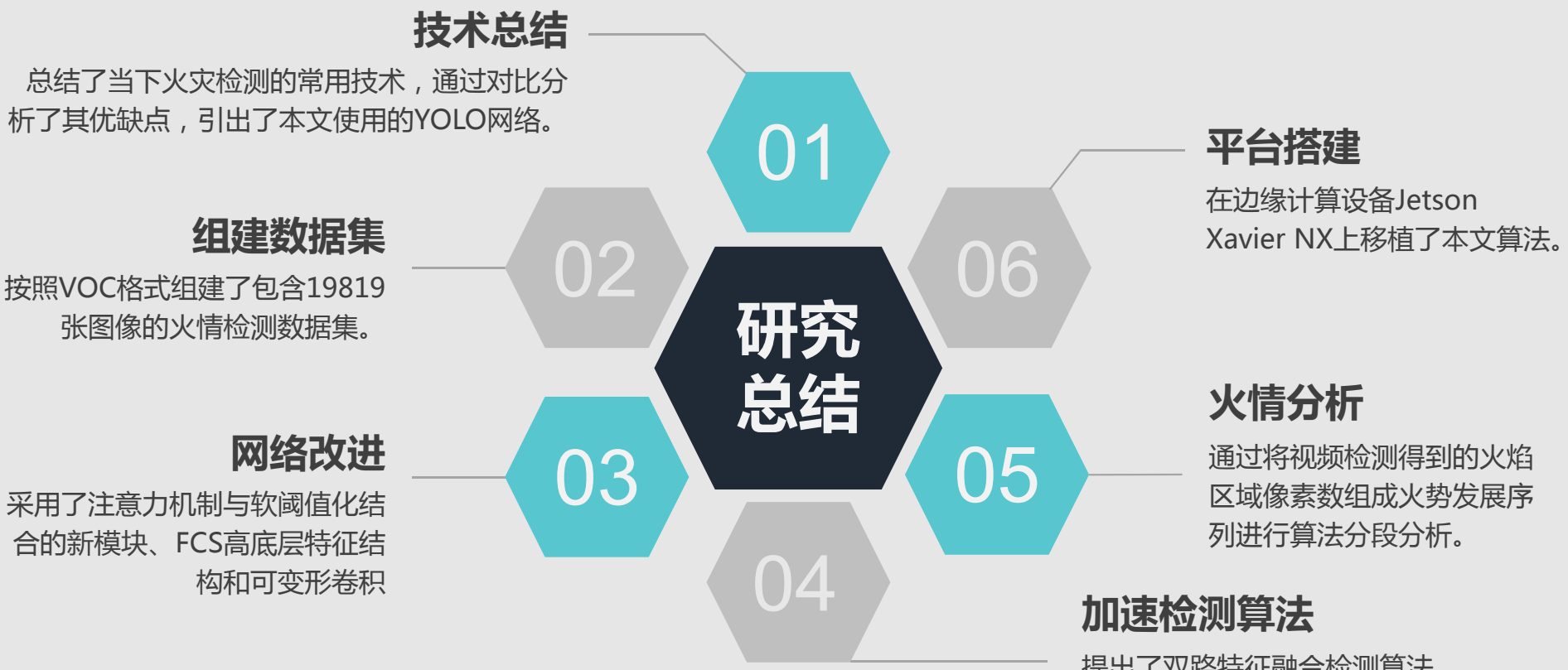


第六部分

总结与展望

- 总结
- 展望

6.1 总结



技术总结

总结了当下火灾检测的常用技术，通过对比分析了其优缺点，引出了本文使用的YOLO网络。

组建数据集

按照VOC格式组建了包含19819张图像的火情检测数据集。

网络改进

采用了注意力机制与软阈值化结合的新模块、FCS高底层特征结构和可变形卷积

平台搭建

在边缘计算设备Jetson Xavier NX上移植了本文算法。

火情分析

通过将视频检测得到的火焰区域像素数组成火势发展序列进行算法分段分析。

加速检测算法

提出了双路特征融合检测算法

6.2 展望

A

数据集扩充

要想卷积神经网络获得更加优秀的性能，还需要进一步增加数据集的数量和标签精度，包括更加广泛的场景。

B

增加类别，降低误检

下一步可以更精细的划分红色、黄色和白色等不同类别的火焰，烟雾也可以继续划分为黄烟、黑烟和白烟等。也可以加入易误检目标如灯光、太阳、云和蒸汽等目标进行训练。

C

火情数据分析

火情分析方面应该还有更多的信息可以提取，也可以在下一步进行火势的在线预测

D

软件平台改进

算法方面，下一步可以利用TensorRT等技术进行进一步硬件加速检测；其次软件方面，还可以再加入用户安全校验功能，防止信息泄露，其软件功能服务也可移植于云端。

研究成果

发明专利

[1] 杜建超, 赵兵兵, **宁阳**, 朱苏雅. 基于深度学习的火情识别定位系统及方法：中国, 202110048216.5[P]. 2021.01.

[2] 杜建超, 韩硕, **宁阳**, 王彬凤. 基于改进的Transformer网络的跌倒检测方法: 中国, 202210256088.8[P]. 2022.03.

学术论文

[1] **宁阳**, 杜建超, 韩硕, 杨传凯.改进DeeplabV3+的火焰分割与火情分析方法[J].西安电子科技大学学报, 2021,48(05): 38-46.（已检索：20214611161685）