



南京大學

NANJING UNIVERSITY

RINC



基于轻量级孪生网络的目标跟踪及应用

- 答辩人：姜少魁 MG1833033
- 导 师：申富饶 教授



目录

CONTENTS

- 1 研究背景
- 2 困难与挑战
- 3 研究内容
 - 基于孪生网络的轻量级目标跟踪算法 (FSSiamese)
 - 融合FSSiamese的视频检测算法
 - 辅助驾驶系统的实际应用
- 4 研究生期间工作成果
- 5 总结及展望

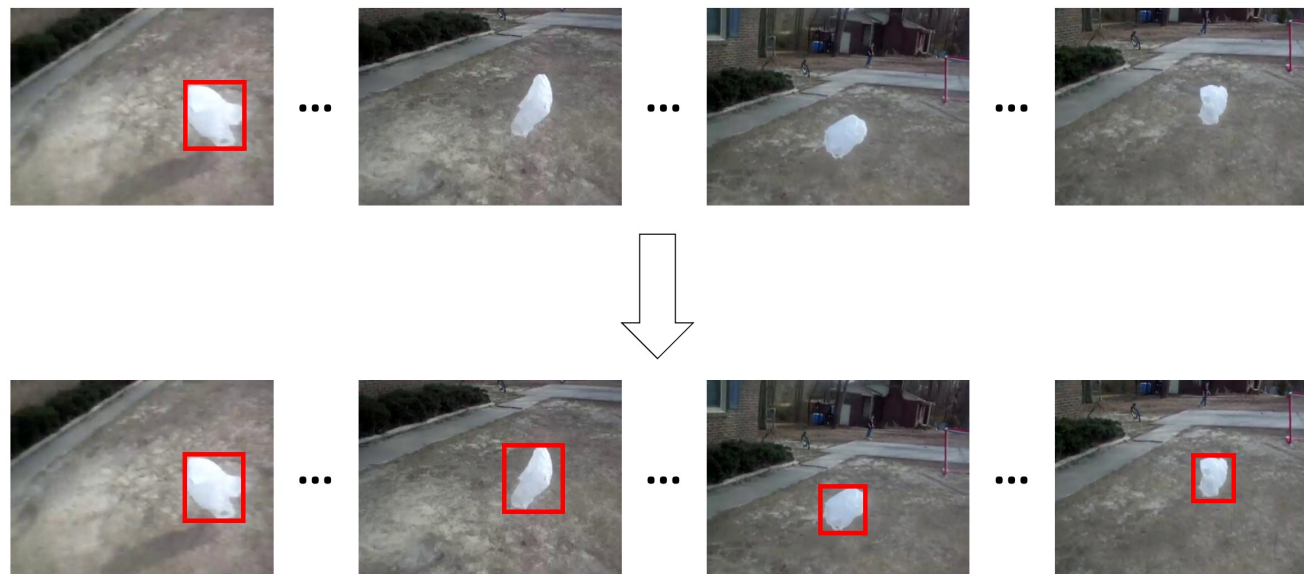


第一部分

Research Background 研究背景



- 单目标跟踪：一个输入为连续图片序列或者视频，给定目标的起始位置，输出为后序每一帧中该目标的位置



单目标跟踪示意图



■ 应用场景



人机交互



视频监控



无人驾驶



第二部分

Problems and Challenges 困难与挑战



■ 生成类算法

- 基于粒子滤波的目标跟踪
- 基于均值偏移的目标跟踪

■ 滤波类算法

- MOSSE算法
- KCF算法

■ 深度学习类算法

- SiamFC算法
- GOTURN算法



一、单目标跟踪算法的局限性

- 场景受限，在实际场景中落地相对较少
- 过于追求准确率而牺牲实用性



二、现有孪生网络方法的局限性

- 数据集庞大
- 结构繁杂
- 运行速度仍有提升空间

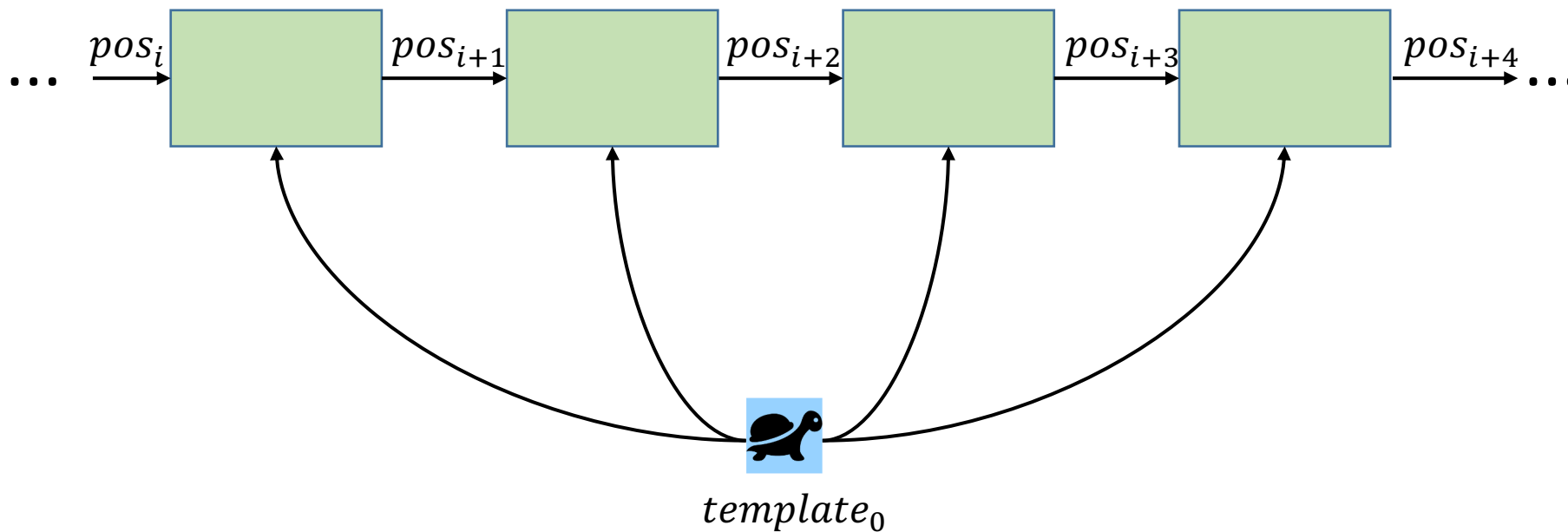


第三部分

Proposed Methods 研究内容

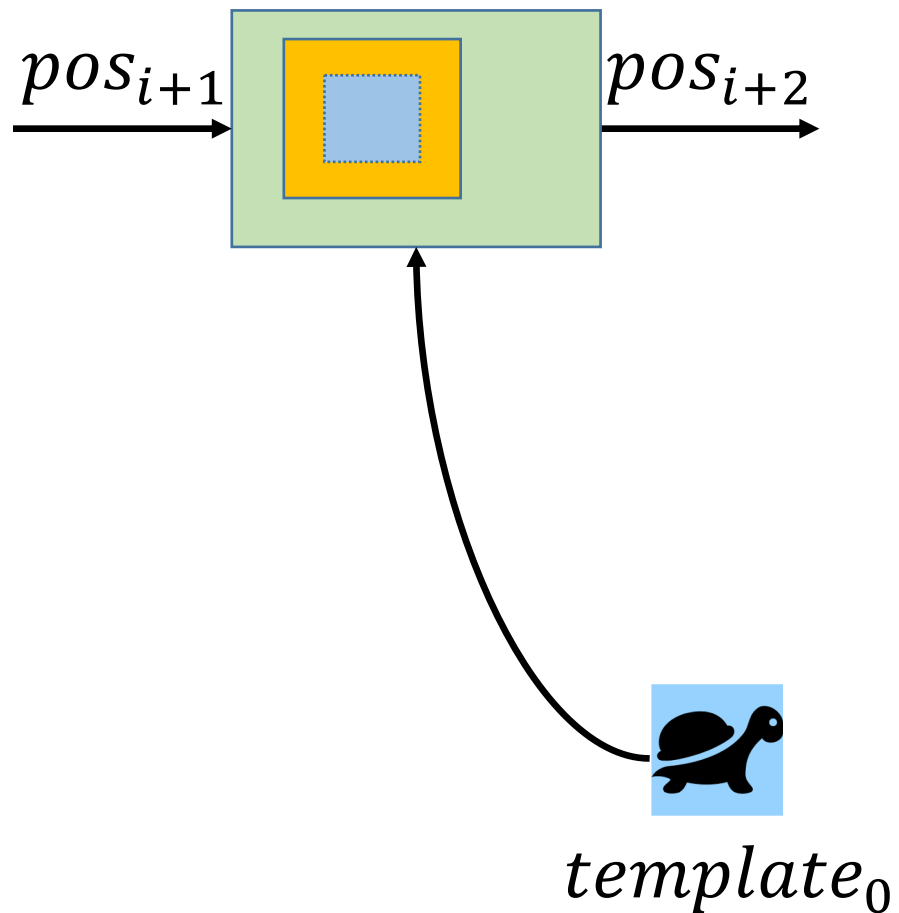


基于孪生网络的轻量级目标跟踪算法FSSiamese



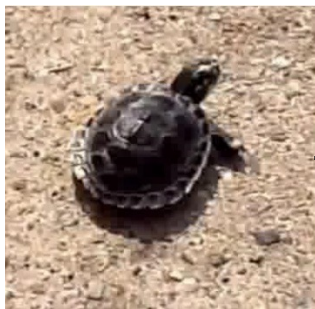


基于孪生网络的轻量级目标跟踪算法FSSiamese

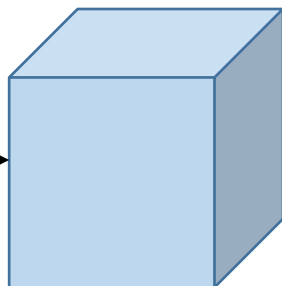




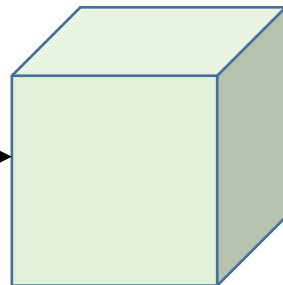
基于孪生网络的轻量级目标跟踪算法FSSiamese



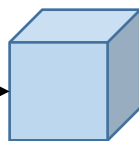
backbone



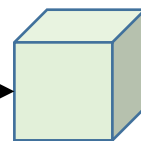
SE_BLOCK



backbone

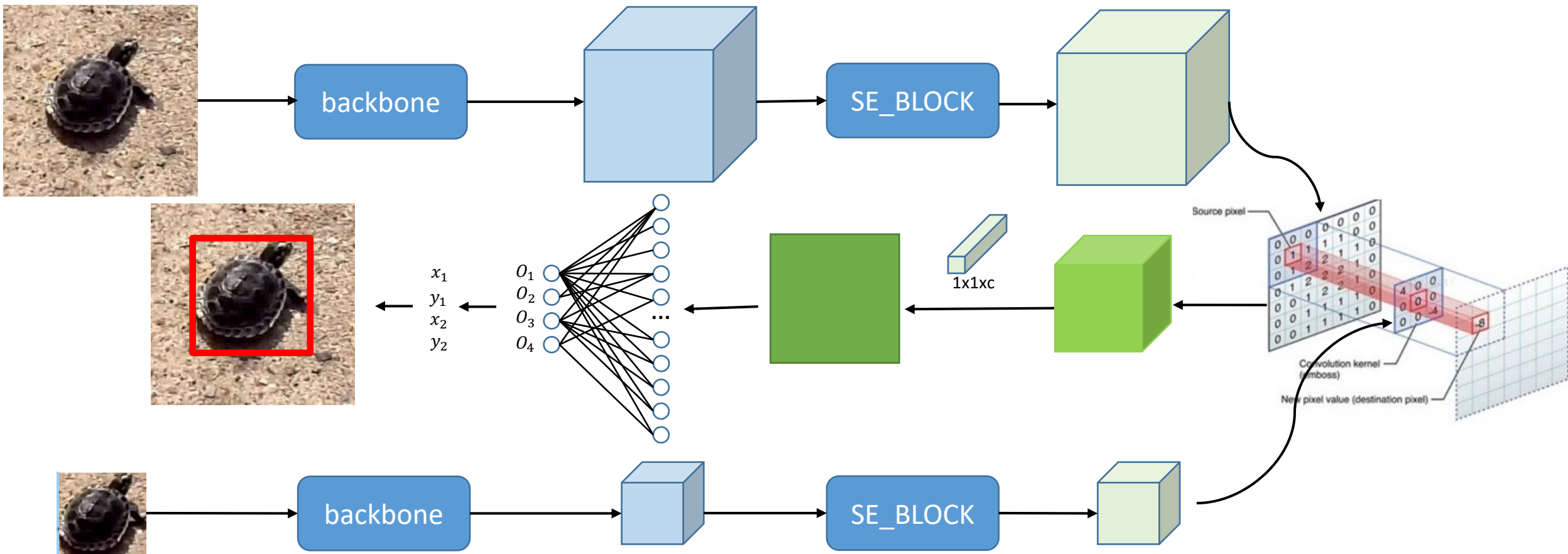


SE_BLOCK



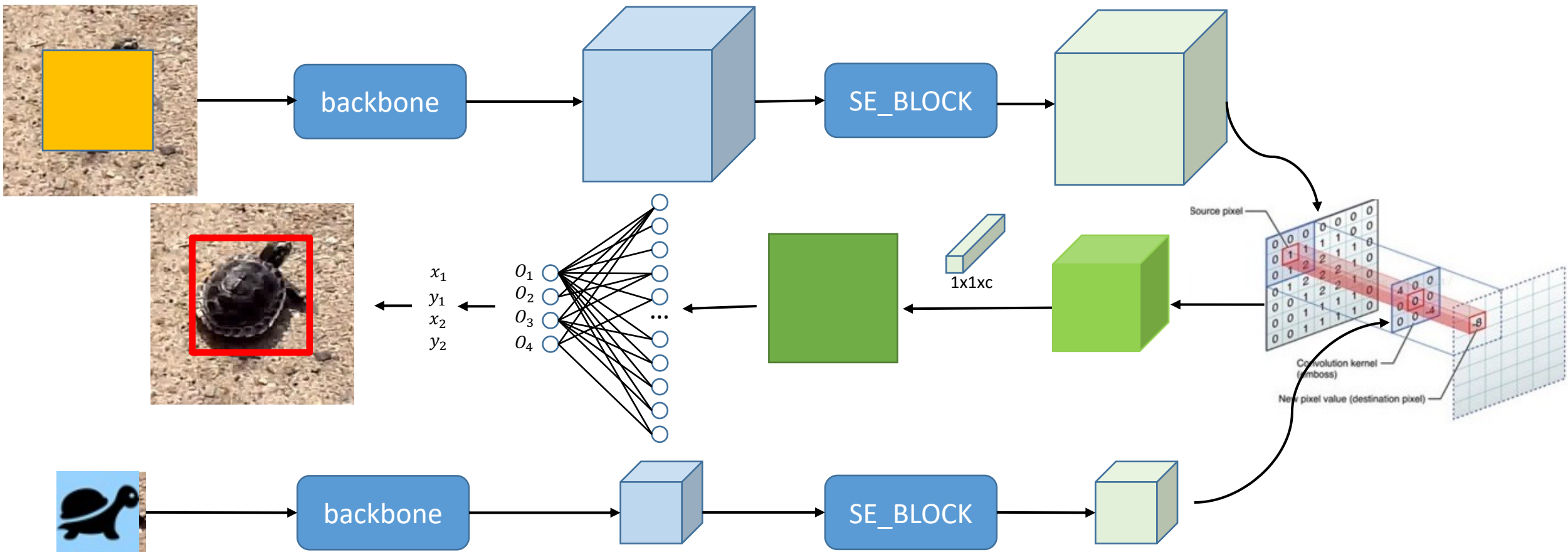


基于孪生网络的轻量级目标跟踪算法FSSiamese



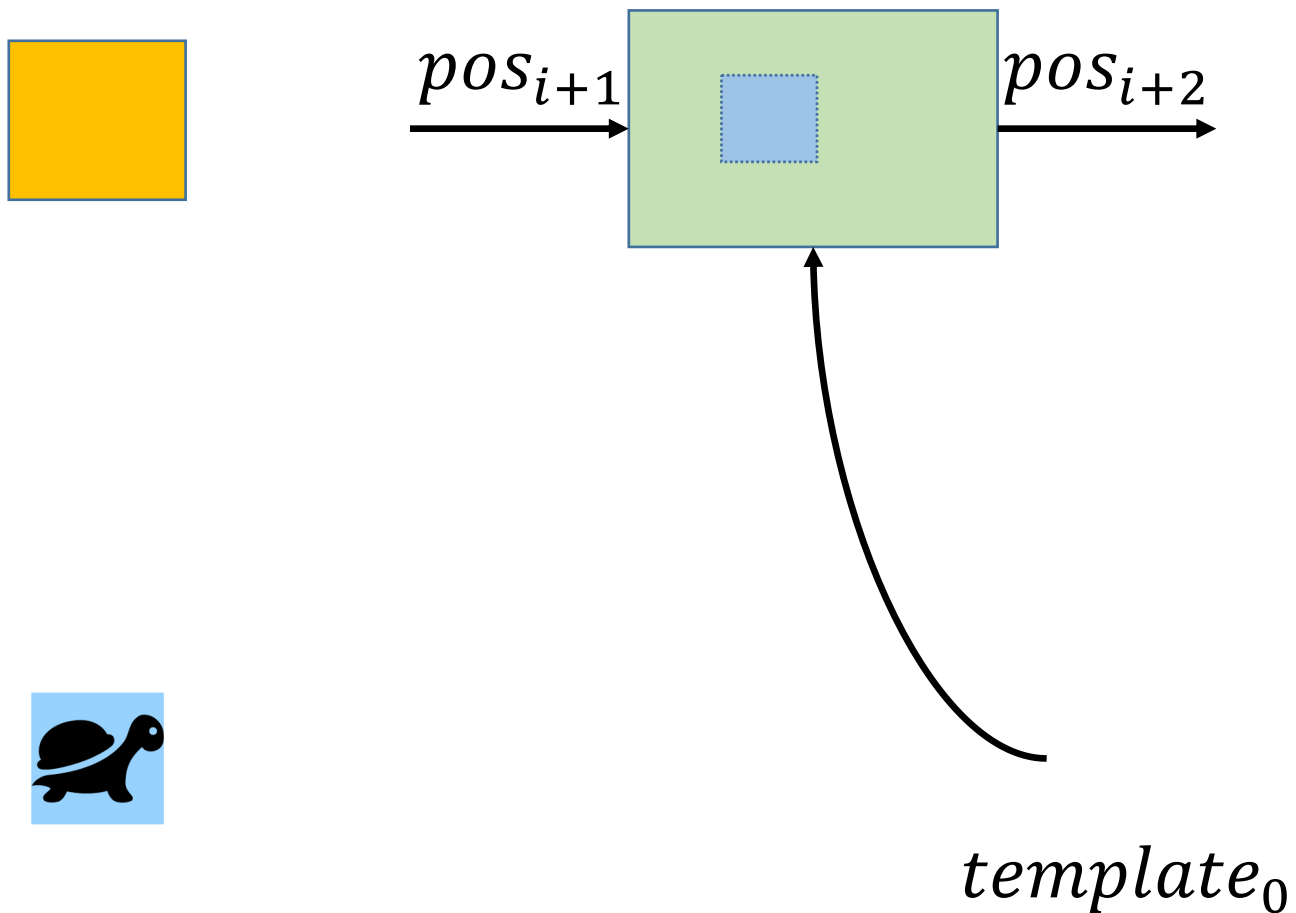


基于孪生网络的轻量级目标跟踪算法FSSiamese



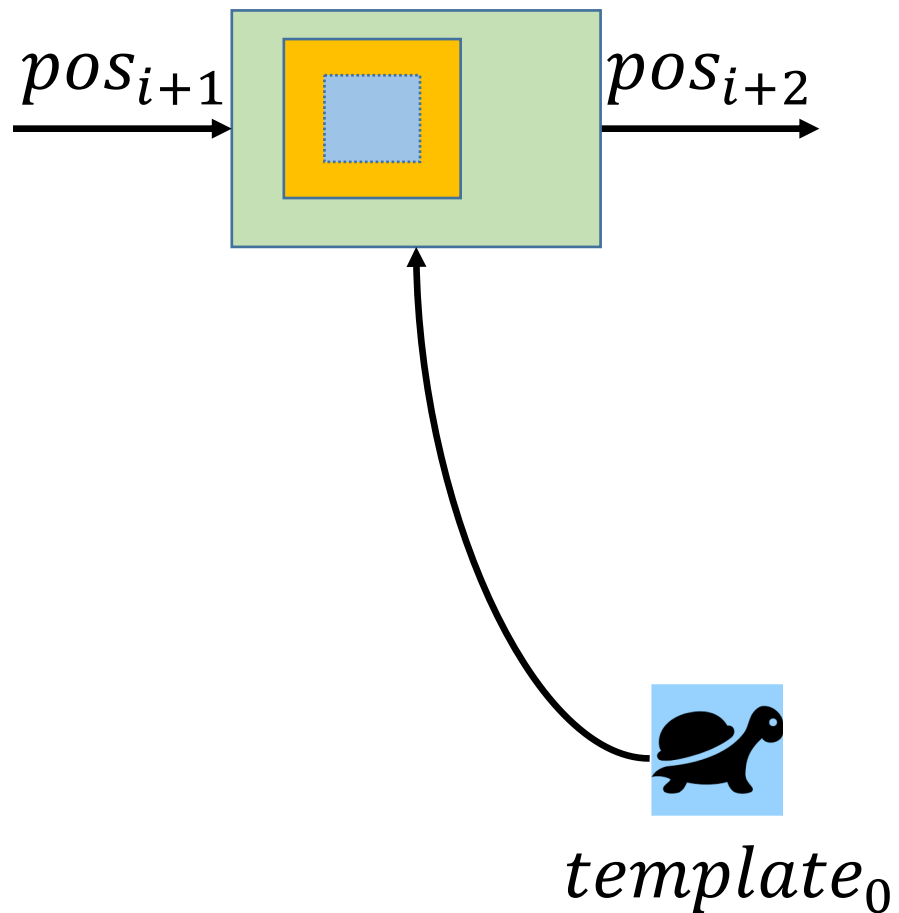


基于孪生网络的轻量级目标跟踪算法FSSiamese



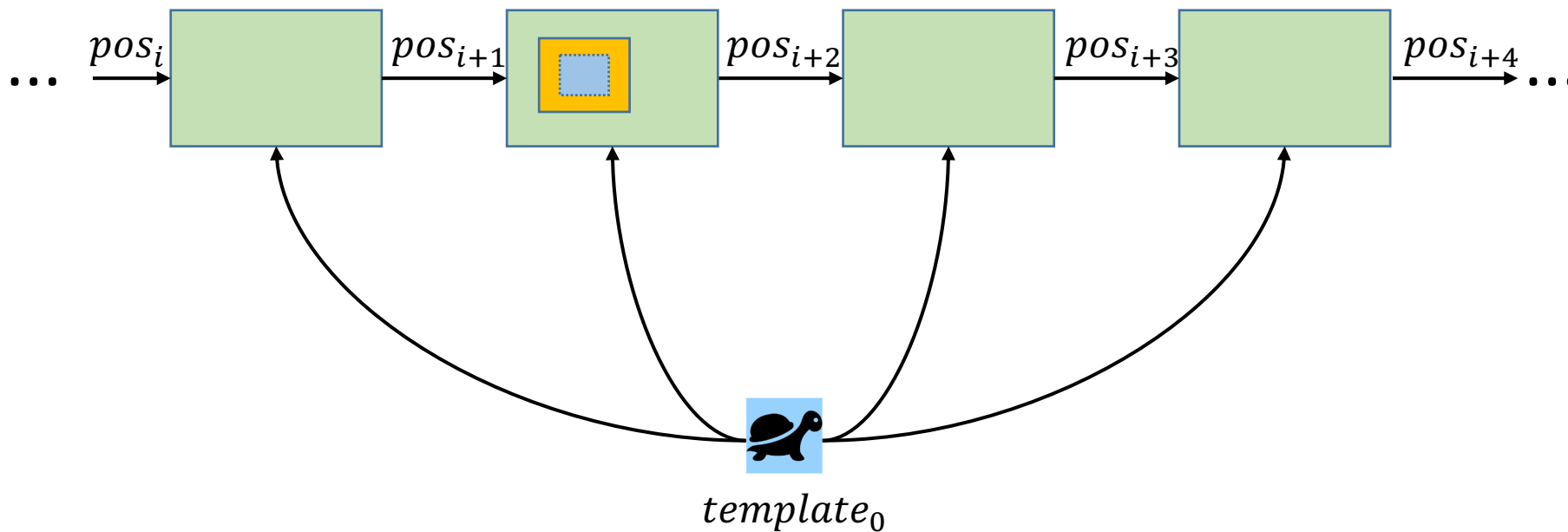


基于孪生网络的轻量级目标跟踪算法FSSiamese



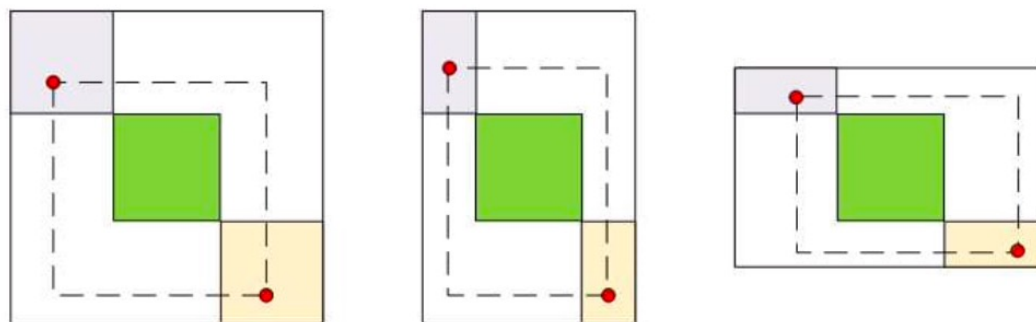
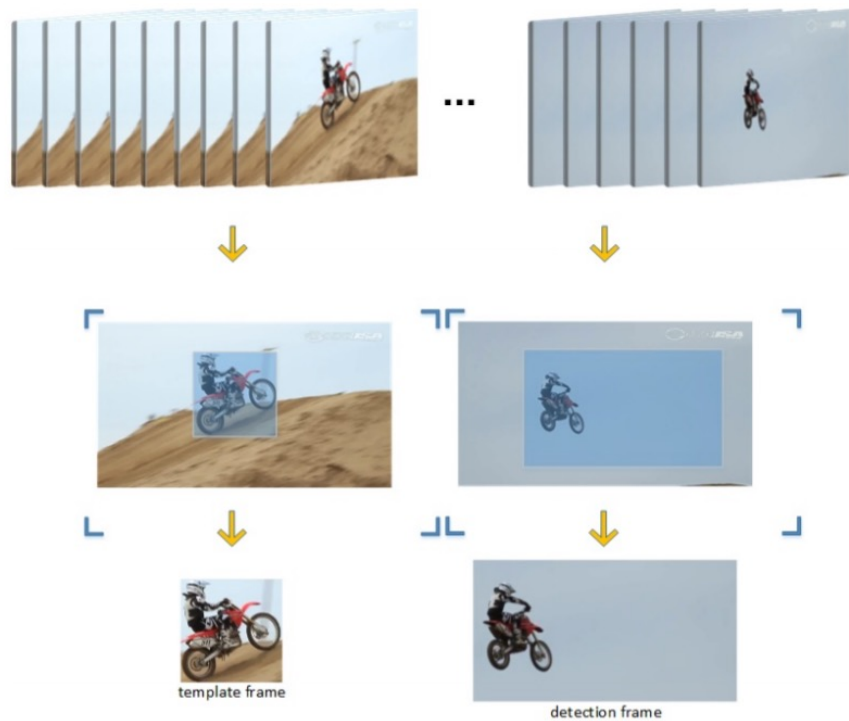


基于孪生网络的轻量级目标跟踪算法FSiamese





模型训练





实验验证

■ VOT跟踪效果对比实验

- 准确率与顶尖的跟踪器相当
- 失败率略高，但仍属于顶尖水平
- 运行速度远超其他跟踪器

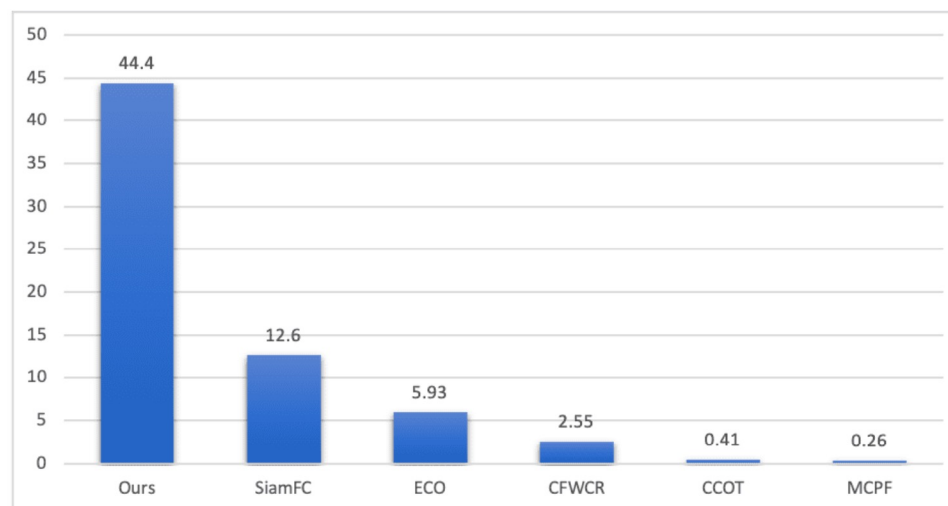
Tracker	Accuracy	Robustness	EAO	EFO
DeepSRDCF	0.56	1.0	0.32	0.38
EBT	0.47	1.02	0.31	1.76
SRDCF	0.56	1.24	0.29	1.99
LDP	0.51	1.84	0.28	4.36
sPST	0.55	1.48	0.28	1.01
SC-EBT	0.55	1.86	0.25	0.80
Ours	0.55	1.95	0.20	52.37

Tracker	Accuracy	Robustness	EAO
CSRDCF++	0.46	0.40	0.21
SiamFC	0.50	0.60	0.18
ECOhc	0.49	0.57	0.18
Staple	0.53	0.69	0.17
ASMS	0.49	0.63	0.17
Ours	0.45	0.81	0.13



实验验证

- VOT2017实时测试速度对比实验
 - 对比了VOT2017中开源并且排名靠前的跟踪器
 - FSSiamese在运行速度上远超其他跟踪器

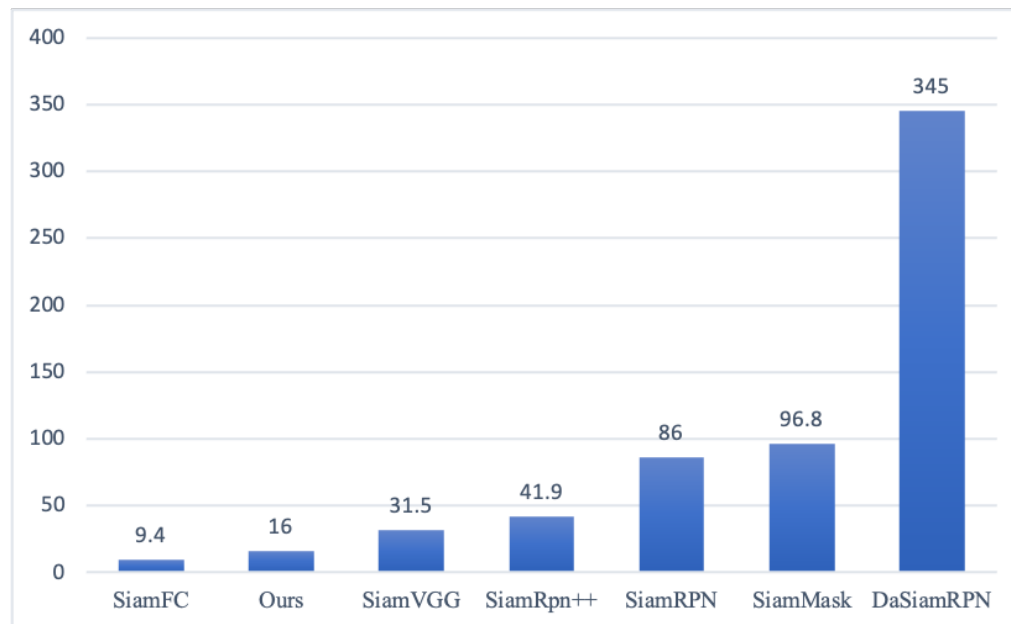




实验验证

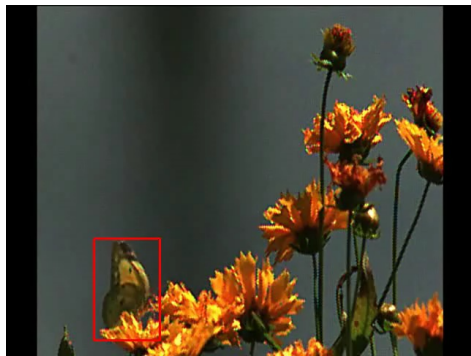
■ 模型大小对比实验

- FSSiamese模型大小与其他孪生网络模型相比足够小
- SiamFC模型大小最少，但需多尺度回归，性能不及FSSiamese





跟踪演示





现有视频检测算法

■ 静态检测算法

- RCNN、CenterNet等
- 缺点：无法充分利用帧间信息，对外观变化、虚焦、遮挡、动态模糊等不敏感

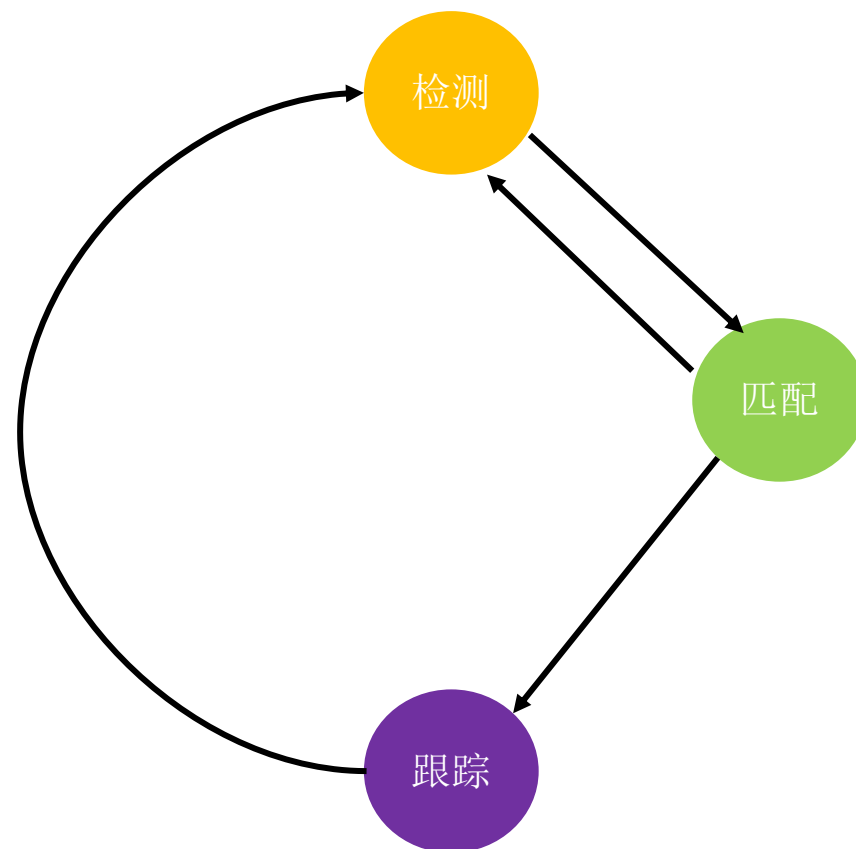
■ 视频检测算法

- T-CNN、FGFA等
- 缺点：网络结构复杂，难以保证实时性、在线运行



引入FSSiamese的视频检测算法

- 检测模块
 - 负责单张静态图像的目标检测
- 匹配模块
 - 负责将前几帧的检测结果进行融合
- 跟踪模块
 - 将前面匹配完成的物体调用FSSiamese跟踪器进行跟踪果返回

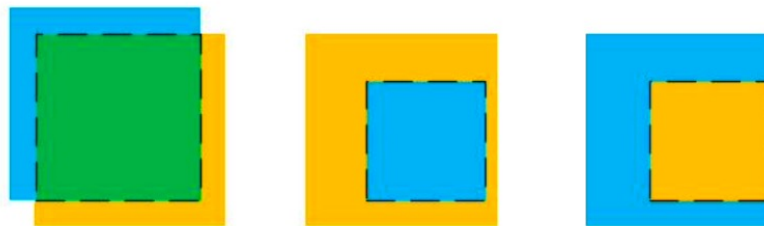


引入FSSiamese的视频检测算法

■ 双阈值匹配算法

- 核心思想：IOU阈值+面积比阈值
- 特征选择：位置特征
- 作用：弥补因物体框尺寸造成的匹配遗漏

$$\phi = \frac{A_t^G \cap A_t^T}{A_t^G \cup A_t^T}$$





实验验证

■ ImageNet-VID数据集测试

- 引入FSSiamese之后，AP值提升明显
- 跟踪效果足够好，在阈值设为0.6的情况下，大部分类别下AP值仍有所提升

模型	未引入FSSiamese			引入FSSiamese		
	0.4(%)	0.5(%)	0.6(%)	0.4(%)	0.5(%)	0.6(%)
飞机	45.66	44.24	42.35	51.23	49.51	47.00
鸟	48.67	45.70	39.62	53.17	49.11	41.09
船	30.69	29.88	28.95	32.17	31.14	25.31
猫	51.68	47.15	42.59	61.86	57.02	49.48
牛	47.74	46.38	41.15	58.34	57.63	53.20
狗	50.69	48.63	44.83	55.07	51.30	46.63
马	39.01	38.73	36.48	44.75	44.60	42.73
摩托	24.29	24.08	22.42	30.93	25.90	20.10
人	14.25	12.50	10.37	17.49	13.89	10.47
羊	54.34	54.26	53.34	63.11	62.92	62.02
全部类别	34.26	32.18	28.99	39.92	36.72	32.12



实验验证

■ GOT数据集测试

- 引入FSSiamese之后，AP值提升明显
- 跟踪效果足够好，在阈值设为0.6的情况下，大部分类别下AP值仍有所提升

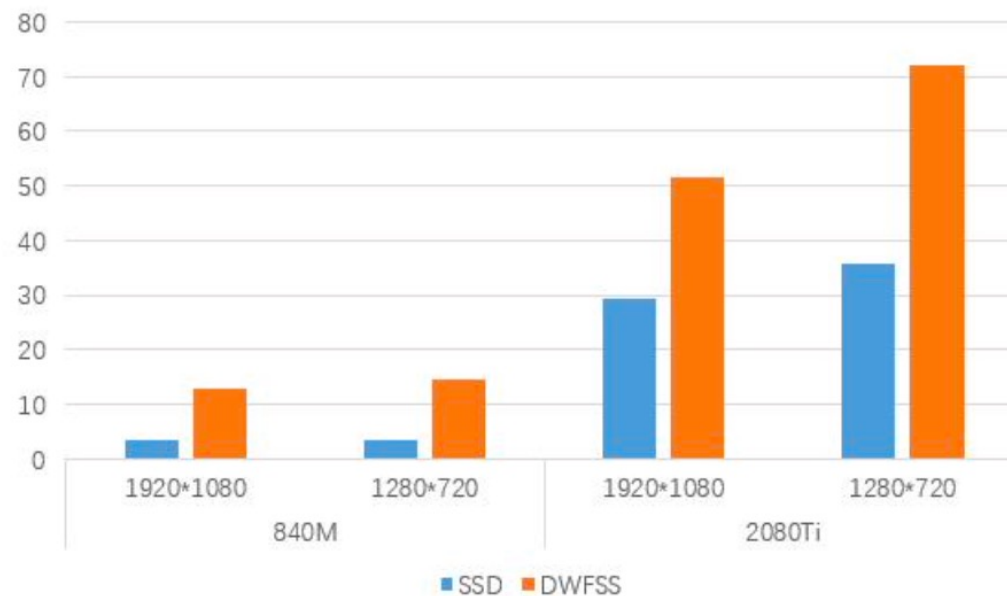
算法	未引入FSSiamese			引入FSSiamese		
	0.4(%)	0.5(%)	0.6(%)	0.4(%)	0.5(%)	0.6(%)
飞机	58.51	58.51	57.99	66.46	64.69	63.63
鸟	34.19	27.99	22.06	44.05	37.56	31.72
船	41.24	40.33	38.40	48.64	48.47	48.21
公交	60.05	56.11	52.94	66.32	64.07	61.66
汽车	19.83	19.36	18.50	19.99	19.89	18.41
猫	45.82	44.83	40.04	62.51	59.70	53.24
牛	19.50	19.23	18.79	19.84	19.56	19.31
狗	31.90	30.92	29.63	40.73	39.73	37.74
马	79.21	78.84	78.23	89.22	87.95	85.92
人	0.39	0.11	0.08	1.04	0.60	0.45
羊	38.66	37.70	34.27	40.89	38.97	35.24
全部类别	24.59	22.67	20.97	29.44	27.67	25.61



实验验证

■ 速度对比实验

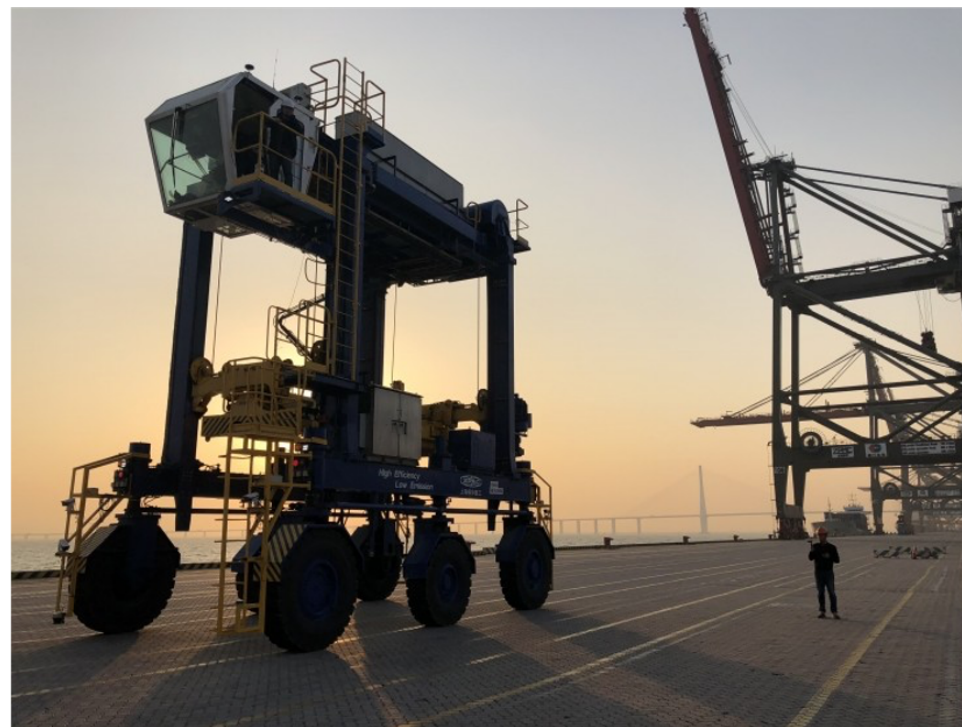
- 在1920*1080和1280*720两种图像尺寸下，速度提升明显
- 不同的硬件环境下，引入FSSiamese后，速度均有所提升





辅助驾驶系统

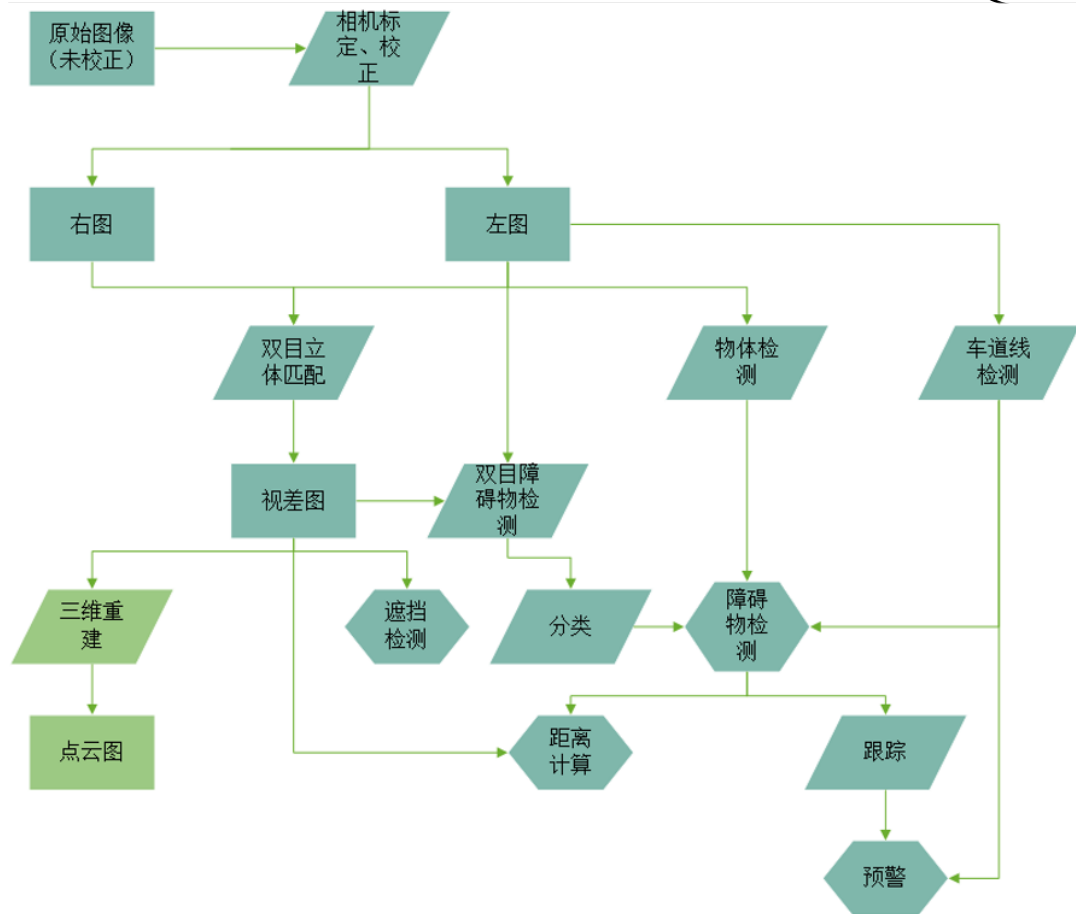
- 辅助驾驶系统任务背景
 - 部署在泰国港口上的轮胎式集装箱门式起重机和轨道式集装箱门式起重机上
- 目的
 - 扫除视觉盲区，辅助驾驶
 - 给司机准确的预警信号





系统架构

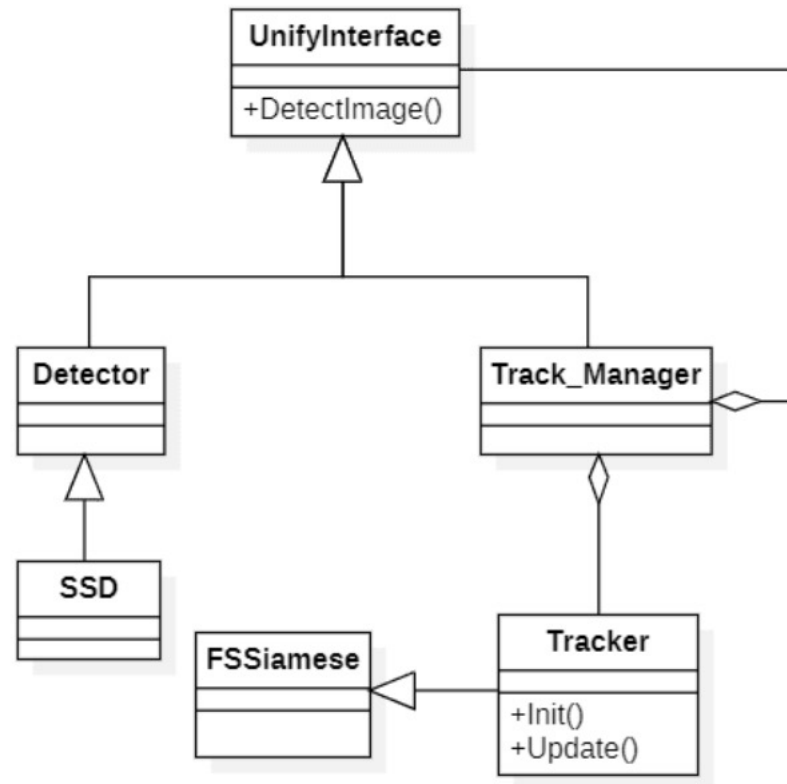
障碍物识别、双目相关、第三目识别三大模块





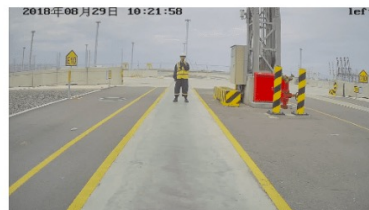
跟踪模块结构设计

- Tracker：定义单目标跟踪的核心函数
 - Init：第一帧指定目标位置
 - Update：输入图像，Tracker返回目标在当前帧的位置
- Track_Manager:负责跟踪相关的逻辑处理

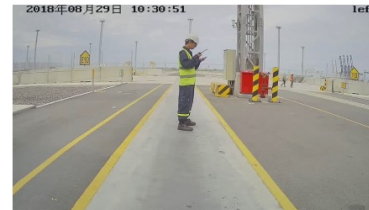


系统数据集测试

- 数据集来源
 - 从辅助驾驶系统应用场景采集的多种姿态下的视频片段，并全部进行了标注
- 实验内容
 - 对比引入FSSiamese模块前后，系统的运行结果



(a) 站立



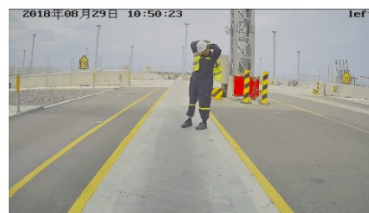
(b) 侧站



(c) 跪



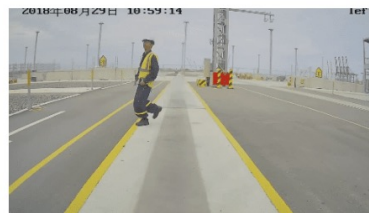
(d) 躺倒



(e) 向前走



(f) 向后走



(g) 向前跑



(h) 向后跑



系统数据集测试

■ 实验对比结果

- 各个阈值下mAP提升大约3个百分点左右

■ 运行时间对比

- 8.9FPS ⇌ 15.2FPS

Model	未引入 $FSSiamese$			引入 $FSSiamese$		
	0.4(%)	0.5(%)	0.6(%)	0.4(%)	0.5(%)	0.6(%)
站立	29.77	29.66	28.34	32.69	32.45	30.94
侧站	10.34	10.20	8.38	13.16	12.10	9.01
跪姿	2.68	2.43	1.42	4.33	4.18	3.41
躺姿	23.64	23.51	22.54	24.79	24.79	24.64
向前走	20.53	17.98	10.21	24.14	22.13	13.95
向后走	22.86	20.30	15.18	31.49	30.00	22.17
向前跑	27.96	23.90	17.98	45.89	41.33	25.70
向后跑	32.68	24.87	12.90	40.26	26.55	15.34
所有类别	23.43	22.80	20.28	26.70	25.92	23.01



系统运行截图

```
jiangshaokui@jiangshaokui-Lenovo-V2000: ~/code/RTG_19/build 68x51
(base) jiangshaokui@jiangshaokui-Lenovo-V2000:~/code/RTG_19/build$ c
make -DUSE_TRACKING=ON ..
-- system arch: x86_64
-- output path: /home/jiangshaokui/code/RTG_19/build/x86_64
-- Found CUDA: /usr/local/cuda-8.0 (found suitable exact version "8.
0")
-- Found CUDA: /usr/local/cuda-8.0 (found version "8.0")
-- Found glog (include: /usr/local/include, library: /usr/local/l
ib/libglog.so)
-- Found OpenCV: /usr/local/include;/usr/local/include/opencv
-- Configuring done
-- Generating done
-- Build files have been written to: /home/jiangshaokui/code/RTG_19/
build
(base) jiangshaokui@jiangshaokui-Lenovo-V2000:~/code/RTG_19/build$ c
make -DUSE_TRACKING=ON .. %& make example
-- system arch: x86_64
-- output path: /home/jiangshaokui/code/RTG_19/build/x86_64
-- Found CUDA: /usr/local/cuda-8.0 (found suitable exact version "8.
0")
-- Found CUDA: /usr/local/cuda-8.0 (found version "8.0")
-- Found glog (include: /usr/local/include, library: /usr/local/l
ib/libglog.so)
-- Found OpenCV: /usr/local/include;/usr/local/include/opencv
-- Configuring done
-- Generating done
-- Build files have been written to: /home/jiangshaokui/code/RTG_19/
build
Scanning dependencies of target common_tools
[ 2%] Building CXX object modules/common/CMakeFiles/common_tools.dir
common_tools.cpp.o
[ 4%] Linking CXX shared library ../x86_64/lib/libcommon_tools.s
o
[ 4%] Built target common_tools
Scanning dependencies of target generate_xml
[ 6%] Building CXX object modules/evaluation/CMakeFiles/generate_xml
.dir/generate_xml.cpp.o
[ 8%] Linking CXX shared library ../x86_64/lib/libgenerate_xml.s
o
[ 8%] Built target generate_xml
Scanning dependencies of target ini_parse
[ 10%] Building CXX object 3rdparty/inifile/CMakeFiles/ini_parse.dir
inifile.cpp.o
```





第四部分

Work and Research Progress 研究生期间工作成果



专利

- 申富饶, 姜少魁, 李俊, 赵健. “一种基于孪生网络的目标跟踪方法” (201910930500.8)

论文

- Shaokui Jiang, Baile Xu, Jian Zhao and Furao Shen " Faster and Simpler Siamese Network for Single Object Tracking" arXiv preprint arXiv:2105.03049 (2021).

比赛获奖

- 韩峰, 李雪健, 赵加成, 姜少魁. 2019首届IKCEST“一带一路”国际大数据竞赛, 国际二等奖



第五部分

Summary and Future 总结及展望



文章总结

基于轻量级孪生网络跟踪算法FSSiamese

- 引入了视觉注意力机制
- 新的回归方式精简了孪生网络结构
- 全新的训练样本生成方法防止过拟合

融合FSSiamese的视频检测算法

- 单目标跟踪与检测的巧妙结合
- 针对简单场景设计的双阈值匹配算法
- 高实时性、在线运行的检测算法

辅助驾驶系统的应用

- 基于计算机视觉和激光的辅助驾驶系统
- 本文提出的跟踪算法作为核心模块引入其中
- 支持模式切换，应对不同的应用场景



未来工作

更优的跟踪网络设计

- 同传统孪生网络跟踪算法用统一的框架实现
- 准确率和速度均表现更优的网络设计
- 通用的网络搜索框架设计

通用的匹配算法

- 适合复杂场景下的匹配算法
- 自适应当前场景，更加灵活地在检测与跟踪间切换

引入更多场景

- 提升单目标跟踪算法的实用性
- 将单目标跟踪应用到更多的实际场景中，不仅仅是论文研究



南京大學
NANJING UNIVERSITY



谢谢！

