

视频目标检测算法库评估系统

使用说明书

单位名称

时间

目 录

1 安装 1

1.1 软硬件环境 1

1.2 安装步骤 1

2 数据集准备 1

3 模型库 1

3.1 MEGA 1

# 安装

## 软硬件环境

Xxx

## 安装步骤

Xxx

# 数据集准备

Xxx

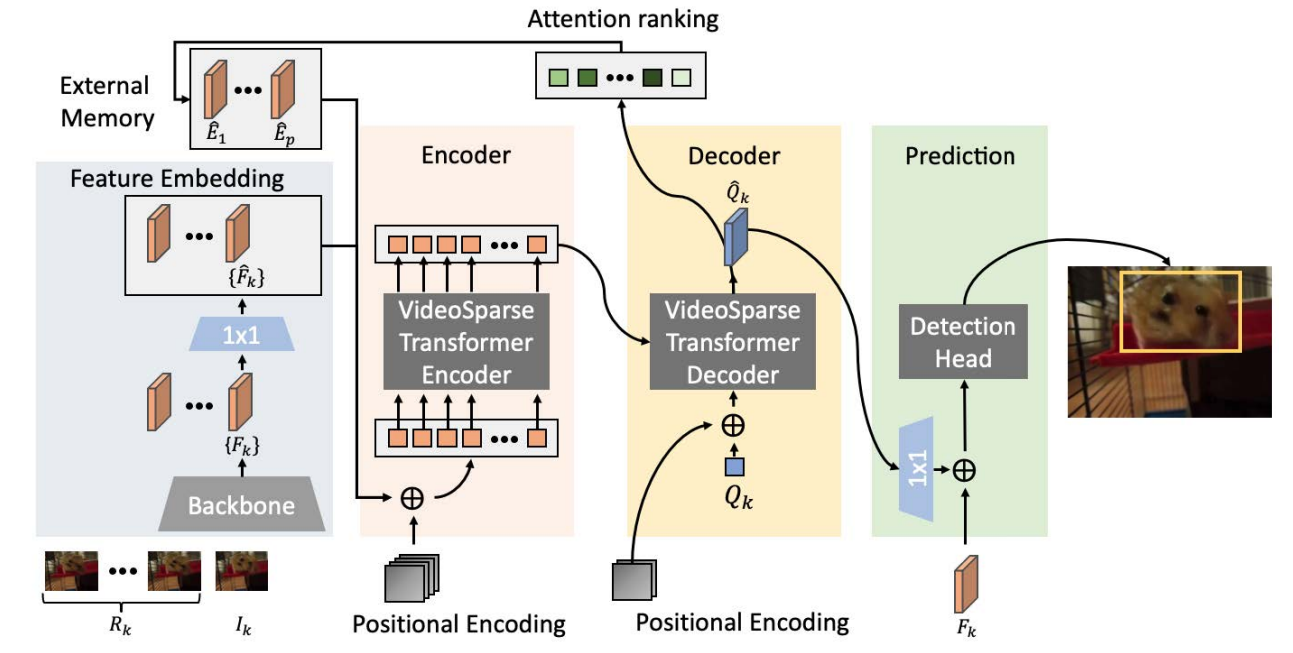
# 模型库

## VSTAM

VSTAM算法来源于论文*Video Sparse Transformer With Attention-Guided Memory for Video Object Detection*。视频对象检测(VOD)是一项具有挑战性的工作，因为随着时间的推移，物体的外观会发生变化，从而可能导致检测错误。最近的研究集中在聚合相邻帧的特征以补偿帧的劣化外观。此外，还提出了使用远帧来处理多帧的劣化现象。由于一个目标的位置可能会在一个遥远的帧显著变化，他们只使用目标候选区域的特征，而不依赖于他们的位置。然而，这些方法依赖于目标候选区域的检测性能，对于劣化外观并不实用。

在目标候选区域检测之前，VSTAM算法对特征元素进行了明智的增强，提出了带有注意引导记忆(VSTAM)的视频稀疏变换。此外，VSTAM算法稀疏地聚合元素智能特征以减少处理时间和内存成本，引入了一种基于聚合的外部内存更新策略，以有效地保存长期信息。算法在ImageNet VID和UA-DETRAC数据集上实现了8.3%和11.1%的基线精度提高。

VSTAM算法在广泛使用的VOD数据集上显示出优越的性能



训练：

python tools/train.py configs/vid/vstam/vstam\_config.py

./tools/dist\_train.sh configs/vid/vstam/vstam\_config.py 8

可选参数

* optimizer: 定义训练时使用的优化算法，这里使用的是随机梯度下降（SGD）算法。学习率为0.01/num\_gpus，动量为0.9，权重衰减为0.0001。
* checkpoint\_config: 配置模型保存的间隔，这里设置每隔1个epoch保存一次模型。
* log\_config: 配置训练过程中的日志记录方式，这里设置每隔50个batch记录一次日志。
* dist\_params: 配置分布式训练的参数，这里使用的是NCCL后端。
* log\_level: 配置日志的详细程度，这里设置为INFO级别。
* load\_from: 模型参数的加载路径，用于继续训练时加载预训练的模型。
* resume\_from: 恢复训练的路径，用于在中断的训练过程中恢复训练。
* workflow: 定义训练过程中的不同阶段和对应的迭代次数。
* lr\_config: 配置学习率的调整方式，预热比例为总迭代次数的1/3，然后在第2和第5个迭代时调整学习率。
* total\_epochs: 总的训练迭代次数。
* evaluation: 定义评估的指标和间隔，这里评估指标为bbox（边界框），每隔7个epoch进行一次评估。
* gpu\_ids: 使用的GPU设备ID范围，用于指定训练时使用的GPU设备。这里设置为使用第0号GPU设备。

测试

python tools/test.py configs/vid/vstam/vstam\_config.py \

--checkpoint checkpoints/$CHECKPOINT\_FILE \

--out results.pkl \

--eval bbox

./tools/dist\_test.sh configs/vid/vstam/vstam\_config.py 8 \

--checkpoint checkpoints/$CHECKPOINT\_FILE \

--out results.pkl \

--eval bbox