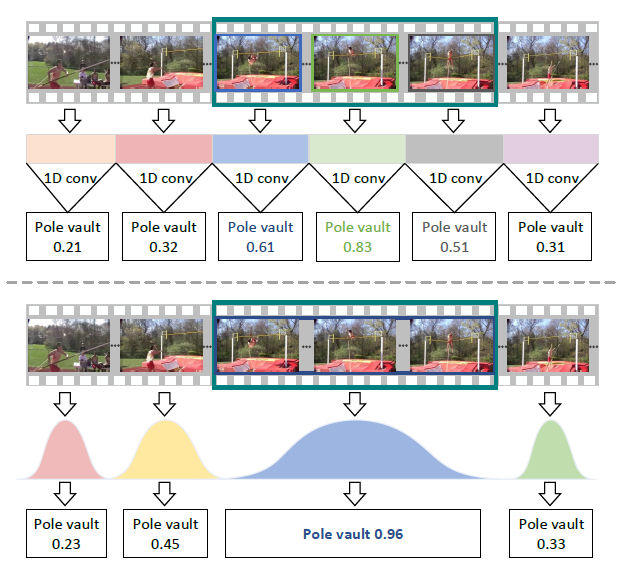




**高斯时间感知网络(GATN)用于动作定位**

## 摘要

*在视频理解中，暂时(temporally)定位视频中的动作是一个基本的挑战。大多数现有的方法往往从图像对象检测中得到启发，并扩展了SSD和更快的R-CNN等先进技术，以在ID序列中动作的暂时定位(temporal locations of action)。然而，由于预先确定的时间尺度(temporal scalse)的设计,结果可能存在鲁棒性(robustness)，这忽略了动作的时间结构，限制了检测复杂变化动作的实用性。在本文中，我们通过提出引入高斯核(Gasssian Kenel)来动态优化每个动作建议的时间尺度来解决这个问题。具体来说，我们提出了高斯时间感知网络(GTAN)-一种新的架构，它新颖的将时间结构的利用集成到一个单阶段的动作定位框架中。从技术上讲，GATN通过学习一组高斯核建模时间结构我们在未剪辑的长视频中解决时间动作定位问题。这一点很重要，因为真实应用程序中的视频通常不受约束，包含多个动作实例以及背景场景或其他活动的视频内容。为了解决这个具有挑战性的问题，我们通过三个基于片段的3D ConvNet，利用深度网络在时间动作定位中的有效性：（1）提议网络识别可能包含动作的长视频中的候选片段；（2） 分类网络学习一对多动作分类模型，作为定位网络的初始化；以及（3）定位网络微调学习的分类网络以定位每个动作实例。我们提出了一个新的损失函数的定位网络显式考虑时间重叠，并实现高的时间定位精度。最后，在预测过程中只使用建议网络和定位网络。在两个大型基准测试中，我们的方法与其他最先进的系统相比取得了显著的优越性能：在Mexaction 2上，mAP从1.7%增加到7.4%，在THUMOS 2014上从15.0%增加到19.0%。*

**

***Figure 1:典型的单阶段动作定位（上）和我们的 GTAN（下）的直觉(intuition)。 典型的方法在每个特征图中固定时间尺度，很少探索动作的时间结构。 相比之下，我们的 GTAN 通过学习一组高斯核来考虑时间结构***

## 介绍

随着在线和个人媒体档案的大量增加，人们正在生成、存储和消费(consuming)大量视频。这一趋势鼓励开发有效和高效的算法来智能解析视频数据(parse video data)。 这些进步成功背后的一个基本挑战是从时间方面(temporal aspect) [6, 9, 17, 30, 39, 43] 和时空方面(spatiotemporal aspect) [11, 18] 对视频进行动作检测。 在这项工作中，主要关注点是时间(temporal action)动作检测/定位，即定位动作开始和结束的确切时间戳，并通过一组类别识别动作。

时间动作定位的一种自然方法是扩展图像对象检测框架，例如 SSD [23] 或 Faster R-CNN [27]，用于在 2D 图像中生成空间边界框以对 1D 序列中的动作进行时间定位[ 4, 19]。 图 1 的上半部分概念化了单阶段动作定位的典型过程。

一般来说，视频序列中的帧级或片段级特征首先聚合为一个特征映射，然后设计多个1D时间卷积层以增加时间感受野的大小并预测动作建议。

然而，每个特征映射中对应于单元的时间尺度是固定的，使得这种方法无法捕捉动作的固有时间结构。因此，在这种情况下，绿框中的一个地面真相行动建议(ground-truth action proposal)被检测为三个。相反，我们建议通过为每个单元学习高斯核来探索行动的时间结构，从而缓解问题，高斯核动态指示行动建议的特定间隔。高斯核的混合甚至可以被分组来描述一个动作，这对于定位不同长度的动作建议更为灵活，如图1底部所示。更重要的是，基于高斯曲线中权重的特征池自然涉及到上下文信息。

通过深入研究动作的时间结构，我们提出了一种新的高斯时间感知网络（GTAN）结构，用于单阶段动作定位。给定一个视频，使用3D ConvNet作为主干来提取片段级特征，这些特征按顺序连接到特征映射中。首先采用两个卷积层和最大池层来缩短特征映射，增加感受野(receptive fields)的时间大小。然后，一系列一维时间卷积层（锚定层）不断缩短特征映射，并输出由每个单元（锚定层）的特征组成的锚定特征映射。在每个锚定层的顶部，为每个单元学习高斯核，以动态预测对应于该单元的动作建议的特定间隔。甚至可以混合多个高斯核来捕获任意长度的行动建议。通过高斯池，每个单元的特征通过聚集上下文单元的特征来升级，这些特征由高斯曲线中的值加权，用于最终行动建议预测。整个架构通过最小化一个分类损失加上两个回归损失（即本地化损失(localization loss)和重叠损失）进行端到端优化。

这项工作的主要贡献是设计了一个单阶段(one-stage )架构GTAN，用于解决视频中的时间动作定位问题。该解决方案还提出了一个优雅的观点，即如何利用动作的时间结构来检测不同长度的动作，以及如何利用上下文信息来促进时间定位，这些都是文献中尚未完全理解的问题。

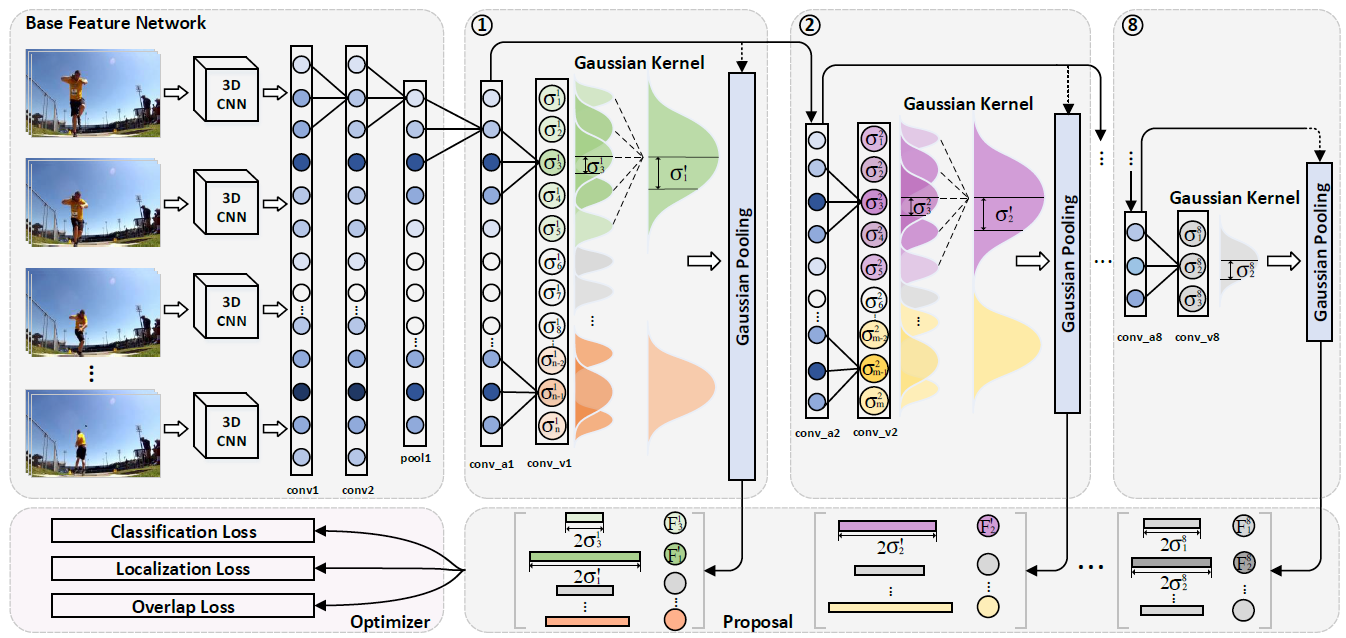
## 相关工作

我们将相关工作简单地分为两类：时间动作提议(temporal action proposal)和时间动作检测(temporal action detection)。前者侧重于研究如何精确定位包含动作的视频片段，而后者则进一步将这些动作分类为已知类。

我们将时间动作提议(temporal action proposal)的方法主要归纳为两个方向：内容无关提议和内容相关提议。与内容无关的提议算法的主流是在视频中均匀或滑动窗口采样[24,33,41]，这会导致进一步分类的大量计算。相比之下，与内容相关的提议方法，例如[3,5,7,8,21]，在培训期间使用行动提议的标签。例如，Escorcia等人[5]利用长短期记忆细胞学习视频序列的适当编码，将其作为一组辨别状态来指示提议分数。虽然该方法避免了运行多尺度滑动窗口，但仍然需要执行重叠滑动窗口，这在视频持续时间较长时不适用。为了解决这个问题，单流时态建议（SST）[3]通过使用基于GRU的循环模型，仅通过一次生成建议，时态单元回归网络（TURN）[8]以金字塔方式构建视频单元，以避免窗口重叠。与上述以固定的多尺度方式生成建议的方法不同，边界敏感网络（BSN）[21]以更灵活的方式基于三条动作曲线来定位动作边界。然而，这种基于动作的方法可能无法定位密集和短动作，因为很难区分曲线中非常接近的起始峰值和结束峰值。

一旦行动建议的本地化完成(localization of action proposals completes)，时间行动检测的自然方式是将建议进一步分类为已知的行动类别，以两阶段的方式进行过程[4、12、29、30、38、43]。然而，提案生成和分类的分离可能会导致次优解决方案。为了进一步促进时间动作检测，最近提出了几种单阶段技术[2,19,40]。例如，单流时间动作检测（SS-TAD）[2]利用基于递归神经网络（RNN）的架构来联合学习动作提议和分类。受SSD[23]的启发，Lin等人[19]设计了一维时间卷积，以生成多个时间动作锚，用于动作建议和检测。此外，随着强化学习的发展，Yaung等人[40]探索RNN，以学习一种以端到端的方式预测行动起点和终点的一瞥策略。然而，由于预定的时间尺度，大多数单阶段方法仍然面临着将所有行动建议本地化的挑战

简而言之，我们的方法属于单阶段时间动作检测技术。与前面提到的通常预先确定动作建议时间尺度的单阶段方法不同，本文中的GTAN不仅研究了通过高斯核学习时间结构，还研究了如何更好地利用上下文信息进行动作定位。



***Figure 2: 我们的高斯时间感知网络（GTAN）架构概述。通过3D ConvNe将输入视频编码为一系列剪辑级特征，这些特征按顺序连接为特征图。两个1D卷积层加上一个最大池层可以增加感受野的时间大小。八个一维卷积层级联生成不同时间分辨率的多个特征图。在每个特征图的顶部，在每个单元上学习高斯核，以预测动作建议的特定间隔。此外，将多个重叠度高的高斯核混合到一个较大的高斯核中，以检测不同长度的长动作。通过高斯 池，行动建议是由高斯曲线中的值加权的上下文单元的特征聚合而成。GTAN与行动分类损失加上两个回归损失共同优化，即每个提案的本地化损失和重叠损失。***

## 高斯时间感知(Temporal Awareness)网络

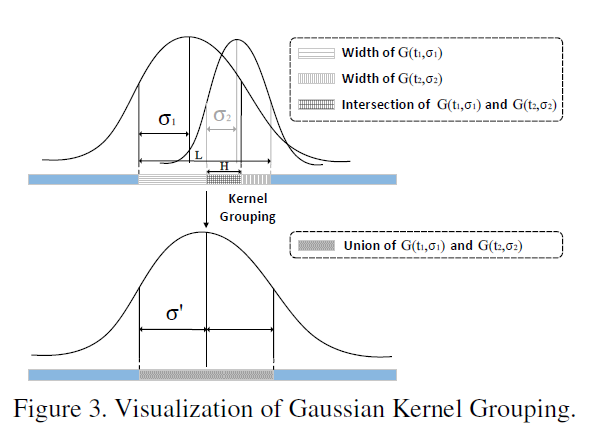
在本节中，我们将详细介绍提议的(proposed)高斯时间感知网络（GTAN）。图 2 说明了我们的动作定位架构的概述。

它由两个主要部分组成：**一个基本特征网络**和**一个具有高斯核的一维时间卷积层级联**。 基本特征网络是从连续视频片段中提取特征图，将其馈入级联的一维卷积层以生成多个不同时间分辨率的特征图。 对于一个特征图中的每个单元，学习一个高斯核来控制与该单元相对应的动作提议的时间尺度，因为训练进行。 此外，设计了一种高斯核分组算法，将具有高重叠的多个高斯核合并到一个更大的核中，以捕获具有任意长度的长动作。 具体来说，每个动作提议都是通过聚合由高斯曲线中的值加权的上下文单元的特征来生成的。 整个网络通过动作分类损失和两个回归损失联合优化，即定位损失和重叠损失，分别用于学习每个动作提议的动作类别标签、默认时间边界调整和重叠置信度得分。

### 基本特征网络

动作定位的最终目标是在时间维度上检测动作实例。给定一个输入视频，我们首先通过3D ConvNet从连续视频中提取视频级别的特征(clip-level features)，该3D ConvNet可以捕获视频的外观和运动信息。具体来说，一系列特征 是从3D ConvNet中提取，其中T是时间长度。我们将所有特征连接到一个特征图中，然后将该图输入两个1D卷积层（“conv1”和“conv2”，时间核大小为3，步幅1）加上一个最大池层（“pool1”，时间核大小为3，步幅2），以增加感受野的时间大小。基本特征网络由3D ConvNet、两个一维卷积层和最大池层组成。进一步利用基本特征网络的输出生成动作建议(action proposal)。

### 高斯核学习



给定基本特征网络输出的特征图，单阶段动作定位的自然方法是叠加1D时间卷积层（锚层），以生成分类和边界回归的建议（锚）。这种在每个锚定层中具有预定时间尺度的结构可以捕获时间间隔与感受野大小一致的动作建议，然而，对不同长度的建议的检测存在困难。该设计限制了将复杂变化的动作定位的实用性

为了解决这个问题，我们引入时态高斯核(temporal Gaussian kernel)来动态控制每个特征映射中提议的时间尺度。特别是，如图2所示，八个1D时间卷积层（锚定层）首先级联，以便以不同的时间分辨率生成动作建议。对于锚定层的特征映射中的每个单元，学习高斯核来预测对应于该单元的动作建议的特定间隔。

形式上，我们将第j卷积层的特征映射表示为，其中和是特征地图的时间长度和特征维度。对于中心位置为t的方案，我们通过高斯核利用其时间尺度。通过特征映射单元上的1D卷积层学习的标准偏差，并通过sigmoid运算将值限制在范围内。

高斯核的权重定义为

(1)

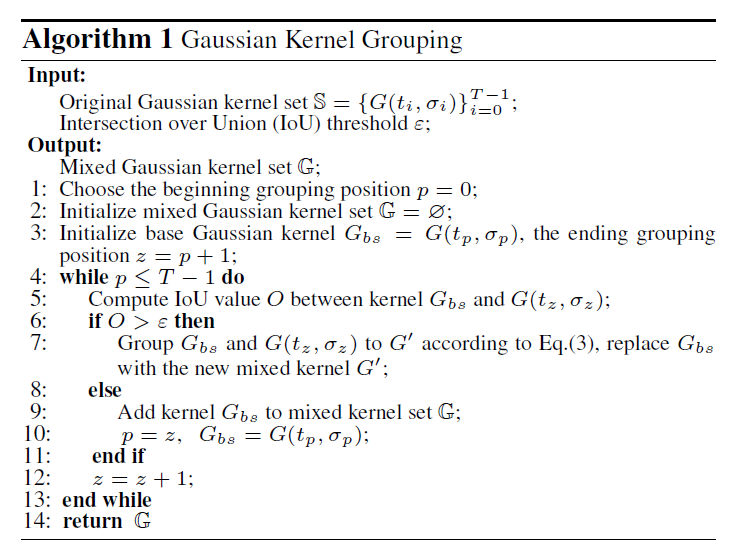
其中Z是标准化常数。基于可以被认为是高斯核中宽度（均方根宽度，RMS）的度量的理论，我们利用作为动作建议的区间度量。具体来说，可以乘以一定的比率来表示默认的时间边界：

(2)

其中和是默认时间边界的中心位置和宽度，表示时间比例。还用于特征聚合(feature aggregation)，具有生成行动建议的池机制，将在第3.4节中详细说明。

与传统的1D卷积锚定层相比，该层在第j层将时间尺度固定为，我们通过利用每个方案的学习高斯核来使用动态时间尺度来探索具有复杂变化的动作实例。

### 高斯核分组



通过学习时间高斯核(temporal Gaussian kernels)，大多数动作实例的时间尺度可以用预测的标准差来表征。然而，如果学习到的高斯核彼此跨越并重叠，则这可能隐含地指示以这些高斯核中的灵活位置为中心的长动作。换句话说，利用这些原始高斯核的中心位置来表示这个长建议可能并不合适。为了缓解这个问题，我们尝试生成一组新的高斯核来预测长期行动方案的中心位置和时间尺度。受[43]中时间动作分组思想的启发，我们针对该目标提出了一种新的高斯核分组算法。

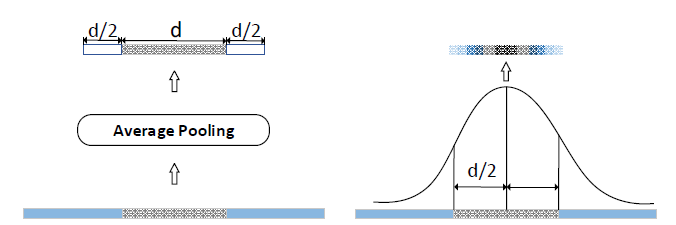
图3说明了时态高斯核(temporal Gaussian Kernel)分组的过程。给定两个相邻的高斯核和，其中心位置和标准偏差为和，我们使用第3.2节中定义的默认时间边界的宽度计算两个核之间的时间交集和并集。在图3的上半部分，两个核之间的时间交集长度为H，而并集长度为L。如果两个内核之间的 Intersection over Union (IoU)

超过某个特定的阈值 ，我们将它们合并为一个高斯核（图 3 的底部）。 新的混合高斯核公式如下

(3)

在每个特征图中，算法 1 详细说明了生成合并内核的分组步骤。

### 高斯池化



***Figure 4 手动扩展加平均池策略（左）和高斯池策略（右）的比较，用于涉及动作提议的时间上下文信息***

使用学习(learned)和混合(mixed)的高斯核，我们根据高斯曲线中的值计算特征图的加权和，得到聚合特征 F。具体来说，给定第 j 层中心位置 t 处的高斯核的加权系数 ，建议的聚合特征表示为

(4)

其中表示被进一步用于动作分类和时间边界回归。

上述高斯池化机制固有地考虑了每个动作提议周围的上下文贡献。 与手动扩展加平均池化策略来捕获视频上下文信息（图 4 的左侧部分）相比，我们提供了一种优雅的替代方案，可以根据重要性自适应地学习加权表示（图 4 的右侧部分）。

### 网络最优化

根据高斯池中每个提议的表示，三个1D卷积层被并行使用，分别预测动作分类分数、定位参数和重叠参数。行动分类得分表示属于C行动类别加上一个“背景”类别的概率。定位参数表示相对于默认中心位置和宽度的时间偏移，它们用于调整时间坐标

(5)

其中，和是提案(proposal)的中心位置和宽度。、用于控制时间偏移的影响。特别是，我们定义了一个重叠参数来表示提案的精确IoU预测，这有利于提案在预测中重新排序

在训练阶段，我们累积来自高斯池的所有建议，并通过预测层生成动作实例。 我们的 GTAN 中的整体训练目标通过整合动作分类损失 和两个回归损失，即定位损失 和重叠损失 来表述为多任务损失：

(6)

其中和是权衡参数。 具体来说，我们通过 softmax 损失测量分类损失 ：

其中指示函数 ，如果 n 等于地面实况动作标签(ground truth action label)c，否则。我们将 表示为该提议的默认时间边界与其对应的最接近地面实况之间的 IoU。 如果这个proposal的大于0.8，我们将其设置为前景样本(foreground sample)。

如果低于 0.3，则将其设置为背景样本(background sample)。在训练期间，前景和背景样本之间的比率设置为 1.0。 定位损失被设计为预测前景提议和提议的最接近地面实况实例之间的平滑 L1 损失 [10] ()，其计算公式为

(8)

其中 和 分别表示提案最近的真实实例的中心位置和宽度。对于重叠损失，我们采用均方误差（MSE）损失对其进行优化，如下所示：

最终，通过惩罚三个损失，以端到端的方式训练整个网络。

### 预测和后处理

在动作定位预测过程中，每个候选动作提议的最终排名分数取决于动作分类分数 和重叠参数。

(10)

给定预测动作实例 具有细化边界 、预测动作标签和排名分数 ，我们采用软非极大值抑制 (soft-NMS) [1] 用于后处理。 在 soft-NMS 的每次迭代中，我们将具有最大排名分数的动作实例表示为 。 其他实例 的排名分数 将根据与 计算的 IoU 降低或不降低：

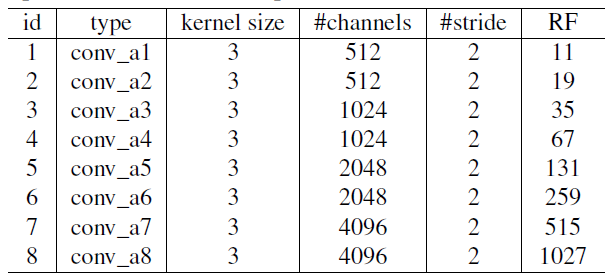
(11)

其中 是衰减参数， 是 NMS 阈值。

## 实验

我们通过在两个流行的视频识别基准，即 **ActivityNet v1.3** [13] 和 **THUMOS14** [16] 上进行时间动作定位实验，凭经验验证了 GTAN 的优点。

### 数据集



***Table 1. 一维时间卷积（锚）层的细节。RF 代表感受野的大小。***

ActivityNet v1.3 数据集包含从 YouTube 收集的 200 个类别中的 19,994 个视频。 数据集分为三个不相交的子集：**训练、验证和测试**，比例2：1：1。数据集中的所有视频都有时间注释。测试集的标签不公开，ActivityNet 数据集上的动作定位性能在验证集上报告。 THUMOS14 数据集有来自 20 个类别的 1,010 个用于验证的视频和 1,574 个用于测试的视频。 在所有视频中，验证集和测试集中分别有 220 个和 212 个带有时间注释的视频。 在[43]之后，我们在验证集上训练模型并在测试集上进行评估。

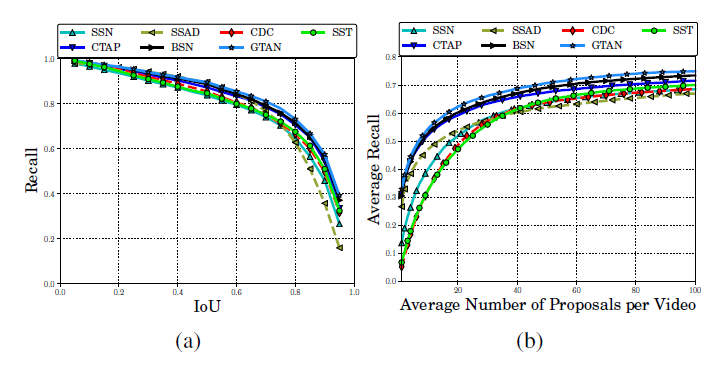
### 实验设置

**实现(Implementations)**。 我们利用 Pseudo-3D [26] 网络作为我们的 3D 主干。 网络输入是一个 16 帧的剪辑，帧的采样率设置为 8。pool5 层的 2,048 路输出被提取为剪辑级特征。表 1 总结了 1D 锚层的结构。此外，我们选择从 [22] 导出的三个时间尺度比。 高斯分组中的IoU阈值 “通过交叉验证设置为0.7。

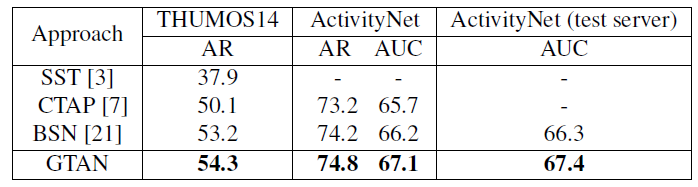
平衡参数和也在验证集上确定并设置为 2.0 和 75。和在 soft-NMS 中设置为 0.8 和 0.75。 通过交叉验证，参数 和 均设置为 1.0。 我们在 Caffe [15] 平台上实现 GTAN。 在所有实验中，我们的网络都是通过使用 0.9 动量的随机梯度下降 (SGD) 来训练的。 初始学习率设置为 0.001，在 THUMOS14 上每 2.5k 次迭代和 ActivityNet 上每 10k 次迭代后降低 10%。 小批量大小为 16，权重衰减参数为 0.0001。

**评估指标。** 我们遵循每个数据集中的官方评估指标进行动作检测任务。 在 ActivityNet v1.3 上，使用 IoU 阈值在 0.5 和 0.95（含）之间且步长为 0.05 的平均精度 (mAP) 值进行比较。 在 THUMOS14 上，测量 IoU 阈值为 0.5 的 mAP。 我们分别评估 ActivityNet v1.3 和 THUMOS14 中前 100 和前 200 的返回提案(returned proposals)的性能

### 对临时行动建议(Temporal Action Proposal)的评估



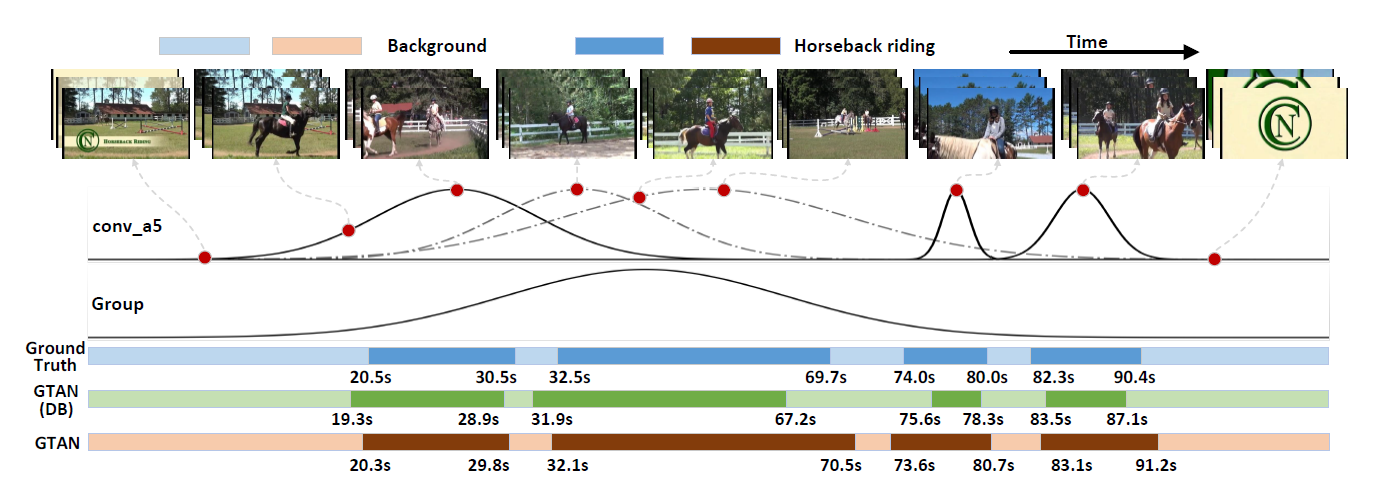
*Figure 5 (a) Recall-IoU 和 (b) ActivityNet 上的 AR-AN 曲线。*



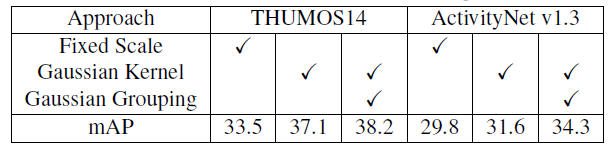
*Table 2 行动建议的 AR 和 AUC 值。 IoU 阈值：THUMOS14 为 [0.5:0.05:1.0]，ActivityNet 为 [0.5:0.05:0.95]*

我们首先考察了临时行动建议任务的性能，即只评估行动建议的边界质量，而不考虑行动类别。我们比较了以下先进的方法：（1）结构段网络（SSN）[43]通过时间动作分组生成动作建议 (2) Single Shot Action Detection (SSAD) [19] 是 Single Shot Detection [23] 的一维变体版本，它通过多个时间锚层生成动作建议。 (3) ConvolutionDe-Convolution Network (CDC) [29] 构建了一个 3D ConvDeconv 网络，以在帧级别精确定位动作实例的边界。 （4）边界敏感网络（BSN）[21]用三个动作曲线定位时间边界，并用神经网络重新排列提案。 （5）单流时间动作提议（SST）[3]构建了一个基于RNN的动作提议网络，可以在长视频序列的单流中实现以产生动作提议。 (6) Complementary Temporal Action Proposal (CTAP) [7] 平衡了滑动窗口和最终行动建议的行动分组方法之间的优缺点。

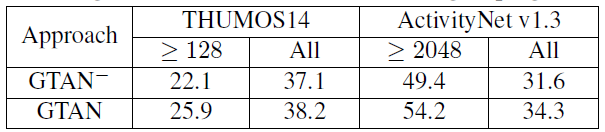
我们对两个数据集的行动建议采用不同IoU（AR）中平均召回率的标准度量。此外，在ActivityNet的官方评估之后，我们在图5中绘制了召回IoU曲线和平均召回率与平均每视频提案数（AR-AN）曲线。除了AR度量，AR-AN曲线下的面积（AUC）也在表2中报告，因为AUC是ActivityNet测试服务器上的度量。总的来说，不同指标和两个数据集的性能一致表明，我们的GTAN相对于基线带来了性能提升。尤其是GTAN在THUMOS14和ActivityNet上的AR分别达到54.3%和74.8%，与最佳竞争对手BSN相比，绝对提高了1.1%和0.6%。在在线ActivityNet测试服务器上进行评估时，GTAN在AUC方面超过BSN 1.1%。结果证明了利用时间结构来定位动作的优势。此外，如图5所示，在不同的IoU中不断获得改进。就AR-AN曲线而言，GTAN在不同数量的顶级回报提案上也表现出更好的表现。即使只有不到10个提案被返回，GTAN仍然表现出明显的改善，这表明GTAN受益于动态优化每个提案时间范围的机制，并且正确的提案排名在前。



***Figure 6 ：GTAN在ActivityNet的视频示例上可视化动作定位。高斯核在“conv\_a5”层的输出上学习。第二个和第三个内核被混合成一个更大的内核。默认框（DB）由高斯核预测。***

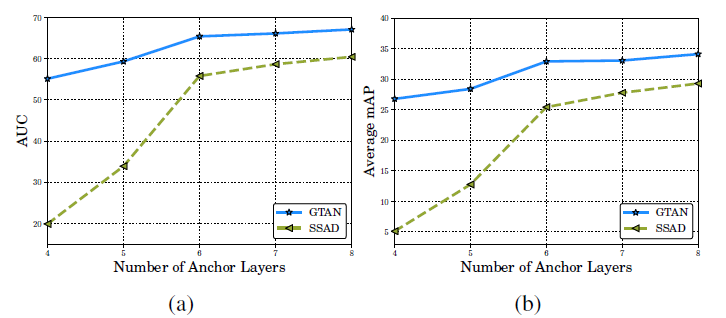


***Table3 GTAN中每种设计的性能贡献。***



***Table4 高斯分组对不同长度动作的评估。GTAN− 不包括GTAN中的高斯分组。***

### 高斯核与分组的评价



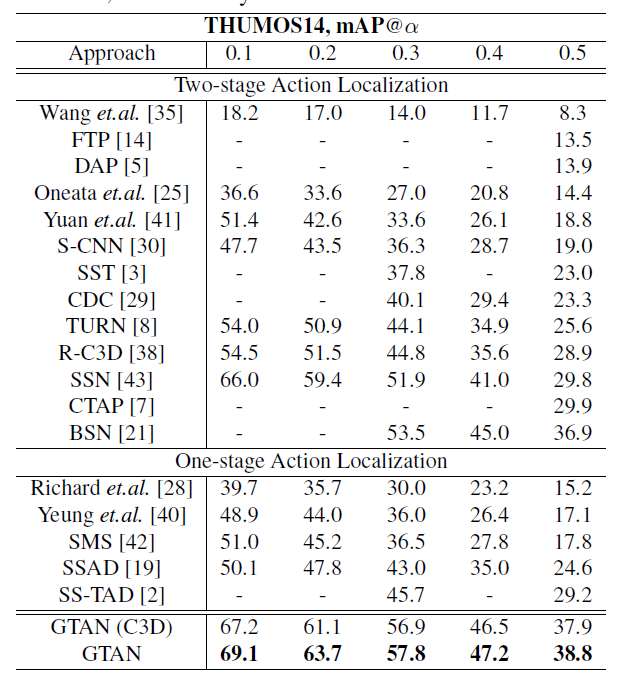
***Figure 7 （a） AUC和（b）在ActivityNet中，不同锚定层数的SSAD和GTAN在时间动作建议和定位任务上的平均mAP性能。***

接下来，我们研究GTAN中的每个设计如何影响时间动作定位任务的整体性能。

固定比例只是为锚定层中的每个单元或锚定采用固定的时间间隔，这种方式在SSAD中采用。高斯核利用为每个锚学习一个高斯核的思想来建模动作的时间结构，并动态预测每个动作建议的时间尺度。高斯分组进一步混合多个高斯核，以表征不同长度的行动建议。在后两种情况下，高斯池被用来用上下文信息增强每个锚的特征。

表 3 通过在两个数据集上考虑 GTAN 中的另一个因素来详细说明 mAP 性能。 Gaussian Kernel 在 THUMOS14 和 ActivityNet v1.3 上分别成功地将 mAP 性能从 33.5% 提高到 37.1% 和从 29.8% 提高到 31.6%。 这在一定程度上揭示了 Fixed Scale 的弱点，其中每个锚点的时间尺度独立于动作提议的时间属性。 相比之下，高斯内核对时间结构进行建模，并动态预测每个锚点的特定间隔。 因此，每个动作提议的时间定位或边界更准确。 此外，通过高斯池化的上下文聚合同时增强了每个动作提议的特征，并导致更好的动作分类。 高斯分组进一步分别贡献了 1.1% 和 2.7% 的 mAP 增长。 结果验证了混合多个高斯核以捕获任意长度的动作提议的有效性和灵活性。 为了更好地验证高斯分组的影响，我们另外评估了 GTAN 对长动作提议的影响。 在这里，我们将 THUMOS14 中超过 128 帧的动作和 ActivityNet v1.3 中超过 2048 帧的动作视为长动作，因为 THUMOS14 中动作实例的平均持续时间为 约 4 秒，远小于 ActivityNet（约50 秒）。表 4 显示了 GTAN 与其变体 GTAN− 之间的 mAP 比较，其中不包括高斯分组。 正如预期的那样，通过涉及高斯分组，在长动作建议上获得了更大程度的改进。

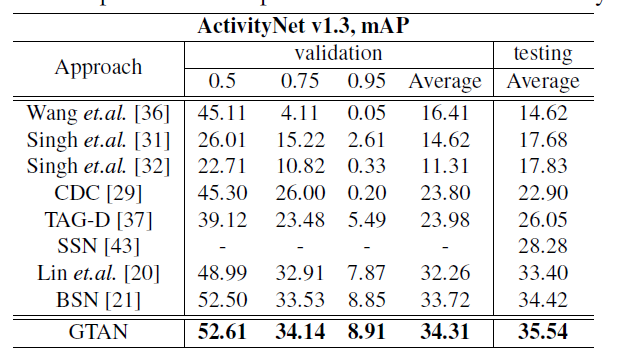
### 锚层数量评估



***Table 5 通过mAP在不同IoU阈值下测量THUMOS14的时间动作检测性能比较。***

在现有的单阶段方法（例如SSAD）中，时间尺度固定在每个锚层中，并且通过增加锚层的数量来实现多个时间尺度的扩展。相反，我们的GTAN学习表5。THUMOS14上时间动作检测的性能比较，通过mAP在不同IoU阈值下测量，每个锚层中的每个锚都有一个高斯核，并动态预测每个锚对应的动作建议的时间尺度。多个高斯核的组合使时间尺度更加灵活。即使有少量锚定层，我们的GTAN在理论上也应该更负责将不同长度的行动建议本地化。图7对SSAD和我们的GTAN在ActivityNet v1上的性能进行了实证比较。3当利用不同数量的锚层时。结果表明，GTAN在4到8个锚定层的不同深度上，在时间动作建议和定位任务上都始终优于SSAD。通常，更多锚定层提供更好的AUC和mAP性能。预计在减少锚固层数量时，SSAD的性能下降幅度比GTAN更大。在4层的极端情况下，GTAN在平均mAP中仍达到26.77%，而SSAD仅达到5.12%，这再次证实了探索时间结构和预测行动建议时间尺度的优势。

### 与先进水平的比较



***Table 6 ActivityNet上时间动作检测的比较***

我们在 THUMOS14 和 ActivityNet v1.3 数据集上比较了几种最先进的技术。 表 5 列出了 THUMOS14 上不同 IoU 阈值的 mAP 性能。 为了公平比较，我们还使用 C3D [34] 作为 3D ConvNet 主干实现了 GTAN。 不同 IoU 值的结果一致表明 GTAN 表现出比其他更好的性能。 特别是 GTAN 的 mAP@0.5 使用 C3D 主干实现了 37.9%，比同样使用 C3D 的一阶段方法 SSAD 和 SS-TAD 分别提高了 13.3% 和 8.7%。 与最先进的两阶段方法 BSN 相比，我们的 GTAN 在 C3D 和 P3D 主干网络上分别带来 1.0% 和 1.9% 的性能提升。 GTAN 的优异结果证明了通过高斯核对动作的时间结构进行建模的优势。

在 ActivityNet v1.3 上，我们在表 6 中总结了验证集和测试集的性能比较。对于测试集，我们将 GTAN 的结果提交到在线 ActivityNet 测试服务器并评估本地化任务的性能。 同样，GTAN 在验证集和测试集上分别超过最佳竞争对手 BSN 0.6% 和 1.1%。 此外，我们的一阶段 GTAN 可能比两阶段解决方案更简单、更快，并且更适用于视频中的动作定位。图 6 展示了来自 ActivityNet 的一个视频的时间定位结果。 在“conv\_a5”层的输出上学习的高斯核和分组也被可视化了。如图所示，高斯核很好地捕捉了每个动作提议的时间结构，并为最终的回归和分类预测准确的默认框。

## 5 结论

我们提出了高斯时间感知网络（GTAN），旨在探索用于时间动作定位的动作的时间结构。 特别是，我们通过学习一组高斯核来动态预测每个动作提议的时间尺度来研究时间结构建模问题。 为了验证我们的主张，我们设计了一个单阶段动作定位框架，它为每个锚层中的每个单元测量一个高斯核。 甚至可以混合多个高斯核，以表示具有不同长度的动作建议。 使用高斯核的另一个优点是通过高斯池化利用上下文信息来增强动作提议的特征，这有利于最终的回归和分类。 在两个视频数据集（即 THUMOS14 和 ActivityNet v1.3）上进行的实验验证了我们的提议和分析。 与一级和二级先进技术相比，也观察到了性能改进。

致谢这项工作得到了中国国家重点研发计划的部分支持，合同号 2017YFB1002203 和 NSFC 号 61872329

## 引用

[1] Navaneeth Bodla, Bharat Singh, Rama Chellappa, and Larry S. Davis. Soft-NMS – Improving Object Detection With One Line of Code. In *ICCV*, 2017.

[2] Shyamal Buch, Victor Escorcia, Bernard Ghanem, Li Fei-Fei, and Juan Carlos Niebles. End-to-End, Single-Stream Temporal Action Detection in Untrimmed Videos. In *BMVC*, 2017.

[3] Shyamal Buch, Victor Escorcia, Chuanqi Shen, Bernard Ghanem, and Juan Carlos Niebles. SST: Single-Stream Temporal Action Proposals. In *CVPR*, 2017.

[4] Yu-Wei Chao, Sudheendra Vijayanarasimhan, Bryan Seybold, David A. Ross, Jia Deng, and Rahul Sukthankar. Rethinking the Faster R-CNN Architecture for Temporal Action Localization. In *CVPR*, 2018.

[5] Victor Escorcia, Fabian Caba Heilbron, Juan Carlos Niebles, and Bernard Ghanem. DAPs: Deep Action Proposals for Action Understanding. In *ECCV*, 2016.

[6] Adrien Gaidon, Zaid Harchaoui, and Cordelia Schmid. Temporal Localization of Actions with Actoms. *IEEE Trans. On* *PAMI*, 35(11):2782–2795, 2013.

[7] Jiyang Gao, Kan Chen, and Ram Nevatia. CFAP: Complementary Temporal Action Proposal Generation. In *ECCV*, 2018.

[8] Jiyang Gao, Zhenheng Yang, Chen Sun, Kan Chen, and Ram Nevatia. TURN TAP: Temporal Unit Regression Network for Temporal Action Proposals. In *ICCV*, 2017.

[9] Roeland De Geest, Efstratios Gavves, Amir Ghodrati, Zhenyang Li, Cees Snoek, and Tinne Tuytelaars. Online

Action Detection. In *ECCV*, 2016.

[10] Ross Girshick. Fast R-CNN. In *ICCV*, 2015.

[11] Georgia Gkioxari and Jitendra Malik. Finding Action Tubes. In *CVPR*, 2015.

[12] Fabian Caba Heilbron, Wayner Barrios, Victor Escorica, and Bernard Ghanem. SCC: Semantic Context Cascade for Efficient Action Detection. In *CVPR*, 2017.

[13] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. ActivityNet: A Large-Scale Video Benchmark for Human Activity Understanding. In *CVPR*, 2015.

[14] Fabian Caba Heilbron, Juan Carlos Niebles, and Bernard Ghanem. Fast Temporal Activity Proposals for Efficient Detection of Human Actions in Untrimmed Videos. In *CVPR*, 2016.

[15] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross B. Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional Architecture for Fast Feature Embedding. *arXiv preprint arXiv:* *1408.5093*, 2014.

[16] Yu-Gang Jiang, Jingen Liu, Amir R.Zamir, and George Toderici. THUMOS challenge: Action recognition with a large number of classes. <http://crcv.ucf.edu/> THUMOS14, 2014.

[17] Colin Lea, Rene Vidal Michael D. Flynn, Austin Reiter, and Gregory D. Hager. Temporal Convolutional Netowrk for Action Segmentation and Detection. In *CVPR*, 2017.

[18] Dong Li, Zhaofan Qiu, Qi Dai, Ting Yao, and Tao Mei. Recurrent Tubelet Proposal and Recognition Networks for Action Detection. In *ECCV*, 2018.

[19] Tianwei Lin, Xu Zhao, and Zheng Shou. Single Shot Temporal Action Detection. In *ACM MM*, 2017.

[20] Tianwei Lin, Xu Zhao, and Zheng Shou. Temporal convolution based action proposal: Submission to activitynet 2017. *arXiv preprint arXiv:1707.06750*, 2017.

[21] Tianwei Lin, Xu Zhao, Haisheng Su, Chongjing Wang, and Ming Yang. BSN: Boundary Sensitive Network for Temporal Action Proposal Generation. In *ECCV*, 2018.

[22] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollar. Focal Loss for Dense Object Detection. In *ICCV*, 2017.

[23] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C. Berg. SSD: Single Shot MultiBox Detector. In *ECCV*, 2016.

[24] Dan Oneata, Jakob Verbeek, and Cordelia Schmid. Action and Event Recognition with Fisher Vectors on a Compact Feature Set. In *ICCV*, 2013.

[25] Dan Oneata, Jakob Verbeek, and Cordelia Schmid. The LEAR submission at Thumos 2014. In *ECCV THUMOS* *Challenge Workshop*, 2014.

[26] Zhaofan Qiu, Ting Yao, and Tao Mei. Learning Spatio- Temporal Representation with Pseudo-3D Residual Networks. In *ICCV*, 2017.

[27] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In *NIPS*, 2015.

[28] Alexander Richard and Juergen Gall. Temporal Action Detection using a Statistical Language Model. In *CVPR*, 2016.

[29] Zheng Shou, Jonathan Chan, Alireza Zareian, Kazuyuki Miyazawa, and Shih-Fu Chang. CDC: Convolutional-De-Convolutional Network for Precise Temporal Action Localization in Untrimmed Videos. In *CVPR*, 2017.

[30] Zheng Shou, DongangWang, and Shih-Fu Chang. Temporal Action Localization in Untrimmed Videos via Multi-stage CNNs. In *CVPR*, 2016.

[31] Bharat Singh, Tim K. Marks, Michael Jones, Oncel Tuzel, and Ming Shao. A Multi-Stream Bi-Directional Recurrent Neural Network for Fine-Grained Action Detection. In *CVPR*, 2016.

[32] Gurkirt Singh and Fabio Cuzzolin. Untrimmed Video Classification for Activity Detection: submission to ActivityNet Challenge. *arXiv preprint arXiv:1607.01979*, 2016.

[33] Kevin Tang, Bangpeng Yao, Li Fei-Fei, and Daphne Koller. Combining the Right Features for Complex Event Recognition. In *ICCV*, 2013.

[34] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning Spatiotemporal Features with 3D Convolutional Networks. In *ICCV*, 2015.

[35] Liming Wang, Yu Qiao, and Xiaoou Tang. Action Recognition and Detection by Combining Motion and Apperance Feature. In *ECCV THUMOS Challenge Workshop*, 2014.

[36] RuxingWang and Dacheng Tao. UTS at activitynet 2016. In *CVPR ActivityNet Challenge Workshop*, 2016.