文章编号:1671-4598(2024)08-0201-08 DOI:10.16526/j.cnki.11-4762/tp.2024.08.029 中图分类号:TP391.4 文献标识码:A

# 基于动态 Transformer 的监控视频摘要系统设计

阮志坚，彭 力
（江南大学 物联网工程学院，江苏 无锡 214000）

**摘要**：监控视频摘要系统是一种重要的技术手段，用于从庞大而复杂的监控视频中提取关键信息，为安全管理和事件分析提供有效支持；随着监控设备的普及和监控视频数据的快速增长，传统的手动摘要方法已经无法满足快速处理和准确提取所需信息的需求，现代的深度学习方法普遍存在计算复杂度高、参数多的问题；针对这一问题，提出了一种基于动态 Transformer 的监控视频摘要模型；自动为每个输入视频帧配置适当数量的 token，通过级联多个 Transformer 模型，并逐渐增加生成的 token 数量，以实现自适应的激活顺序；一旦产生足够置信的预测，推理过程就会终止，并采用了特征重用和注意力重用技术以减少冗余计算；该模型在降低计算复杂度方面取得了显著进展，经实验测试，相较于传统模型，该动态 Transformer 模型在准确率上有所提升，在这两个公开数据集上 $F$ 分数指标分别提高了 3.7%和 0.9%，同时计算复杂度降低了 40%，可以满足精度要求和监控要求，证明模型具有良好的泛化性。

**关键词**：视频摘要技术；动态 Transformer；计算复杂度；特征重用；注意力重用

# Design of Surveillance Video Summarization System Based on Dynamic Transformer

RUAN Zhijian, PENG Li
(School of Internet of Things Engineering, Jiangnan University, Wuxi 214000, China)

**Abstract**: A surveillance video summarization system is an important technical tool, it is used to extract key information from large and complex surveillance videos, and provides an effective support for security management and event analysis. With the popularization of surveillance devices and rapid growth of surveillance video data, traditional manual summarization methods cannot meet the demands of fast processing and accurate extraction of required information. Modern deep learning methods widely have the shortages of high computational complexity and large parameters. To address this issue, a dynamic Transformer-based surveillance video summarization model is proposed. The model automatically assigns appropriate tokens to each input video frame, cascades multiple Transformer models, and gradually increases the number of generated tokens to achieve the adaptive activation order. Once, it generates the sufficient confident predictions, the inference process will terminate. The model adopts the feature reuse and attention reuse techniques to reduce the redundant computations. It makes a significant progress in reducing the computational complexity. Experimental tests show that compared with traditional models, the dynamic Transformer model increases the accuracy, the F score indicators by 3.7% and 0.9% on two publicly available datasets, respectively. At the same time, the computational complexity is reduced by 40%. This model can meet the requirements of precision and surveillance, demonstrating a good generalization performance.

**Keywords**: video summarization techniques; dynamic Transformer; computational complexity; feature reuse; attention reuse

## 0 引言

最近几十年，由于越来越多的人参与到视频的制作和分享过程中来，造成每天有海量的视频被上传到互联网上，这使得互联网上的视频数量呈现指数级的增长趋势。但同时也带来了许多急需解决的问题，比如在视频存储、传播和检索等方面给使用者带来了极大的压力和困难。为了克服上面提出的视频检索困难、传播效率低等问题，视频摘要技术应运而生，并逐渐引起了越来越多人的关注。

监控领域通常需要处理大量的监控视频数据，而视频摘要可以帮助从这些海量视频数据中提取出关键信息，实现对监控场景的快速浏览和分析。通过生成视频摘要，监控人员可以在较短的时间内了解监控画面中发生的重要事件，快速定位关键内容，提高监控效率。通过对监控视频进行摘要生成，可以自动提取出具有代表性的视频片段或关键帧，为后续的视频内容分析和识别提供有效的输入数据。基于视频摘要的监控系统可以更快速地检测到异常事件或关键目标，提高监控系统的实时性和准确性。

**收稿日期**:2024-01-16； **修回日期**:2024-02-23。
**作者简介**:阮志坚(1999-),男,硕士研究生。
**通讯作者**:彭 力(1967-),男,博士,教授,博士生导师。
**引用格式**:阮志坚,彭 力.基于动态 Transformer 的监控视频摘要系统设计[J].计算机测量与控制,2024,32(8):201-208.

视频监控的应用场景多为实时或在线处理。对于这样的场景，算法需要能够及时地对视频监控内容进行处理，并输出相应的摘要结果。视频数据通常具有大规模、高维度、复杂性强等特点，需要消耗大量的计算资源和时间。因此，对于视频摘要算法来说，计算复杂度的控制和优化是非常关键的。

传统的簇类[1]方法包括 VSUMM[2]框架、稀疏字典[3]选择问题和 DR-DSN[4]决策过程，存在难以处理长视频、缺乏时序连贯性的问题。故现有的监控领域用的视频摘要算法主要基于深度学习，其中，长短期记忆网络（LSTM，long short-term memory）[5]被广泛应用，在生成视频摘要的任务中效果显著。利用 LSTM 网络对视频帧进行分类和聚类，从而实现视频摘要的目的。包括 dppLSTM[6]，使用双向 LSTM 来建模视频中的时间依赖关系。还有一些工作将 LSTM 与卷积神经网络（CNN，convolutional neural network）结合起来，对视频进行特征提取和理解，但是 LSTM 是一种循环神经网络（RNN，recurrent neural network）的改进算法，在 LSTM 中，时间步通常对应着序列数据中的每个时间点，而视频摘要中的视频帧则是视频序列中的各个帧。在视频摘要任务中，可以将每个视频帧视为一个时间步的输入，利用 LSTM 模型来学习视频帧之间的时间序列信息，从而生成视频摘要或进行其他相关任务，但无法并行计算，且不能灵活构建帧之间的关系。SUM-FCN[7]不使用 LSTM 来建模视频，而是通过一维全卷积网络（FCN，fully convolutional network）来捕获视频内的长程依赖关系，通过一系列的卷积和池化层，随着网络深入，有效的上下文范围不断增大。考虑到循环模型的缺陷，VASNet[8]引入了注意机制来捕获视频帧之间的全局依赖关系。除了考虑时间依赖关系外，还使用三维卷积神经网络（3D-CNN[9]）来提取视频的时空特征，并通过循环网络进一步建模时间依赖关系。

SumGraph[10]是一种基于图的视频摘要方法，利用递归图建模网络对帧之间的全局依赖关系进行建模。近年来，随着神经网络的发展和大规模视频数据集的蓬勃发展，为了解决这个问题，基于 Transformer 的视频摘要算法逐渐受到关注。这种算法首先使用卷积神经网络（CNN）和长短时记忆网络（LSTM）等对视频进行特征提取，然后使用 Transformer[11]模型对这些特征进行编码，最后通过解码器生成摘要。在这个过程中，不同的视频片段被赋予不同的权重，以反映其在生成摘要过程中的重要性。所以现在很多人使用 Transformer 来代替 RNN。Transformer 通过多头关注机制捕获序列中的全局依赖性，该机制并行编码所有时间步骤。不过，使用 Transformer 来应用视频摘要算法计算复杂度较高：这是由于 Transformer 中引入了 self-attention 机制，需要计算大量的注意力权重。

为了解决这个问题，研究提出了一种基于动态 Transformer 的模型，采用了特征重用和注意力重用技术，以减少冗余计算，该模型可以降低视频摘要算法的计算复杂度。实验结果表明，该模型相较于传统的 Transformer 模型，在准确率方面有所提升，且计算复杂度降低了一定程度。

## 1 系统结构及体系

### 1.1 硬件实现方案

#### 1.1.1 摄像头

本系统配备海康 DS-2CD7A47FWD-XZS 监控摄像头，以实时获取监控视频数据。该摄像头能够实现在 1 920×1 080 分辨率下拍摄 30 fps 的视频，摄像头详细参数如表 1 所示。

表 1 监控摄像头详细参数

| 参数名 | 参数值 |
|---|---|
| 型号 | DS-2CD7A47FWD-XZS |
| 像素 | 400 万 dpi |
| 分辨率 | 1 920 * 1 080 |
| 最大帧率 | 30 fps |
| 信噪比 | 60 dB |

#### 1.1.2 服务器

监控视频摘要系统需要处理大量的视频和图像数据，并进行复杂的计算，包括特征提取、相似性度量等。因此，服务器的选型对于监控视频摘要系统至关重要。本系统的服务器配置如表 2 所示。

表 2 服务器配置

| 类型 | 配置 |
|---|---|
| CPU | E5-2678 v3 |
| GPU | 1 080 Ti * 4 |
| 内存 | 128 G |
| 硬盘 | 2.0 T |

### 1.2 系统软件实现方案

监控视频摘要系统采用 Vue 和 Flask 等前后端开发技术，并使用 MySQL 作为后端数据库，监控视频摘要系统的软件实现方案如图 1 所示。相比于原生的 PyQt 实现方案，Vue 和 Flask 具有更好的前后端分离能力，使得前后端的开发更加独立和高效。同时，Vue 作为一种流行的 JavaScript 库，可以实现美观的用户界面和高效前端逻辑，具有良好的组件化能力和跨平台能力。而 Flask 作为一个轻量级的 Python Web 框架，能够快速搭建后端服务和 API 接口，同时拥有丰富的扩展和插件，使得系统的开发更加便捷。系统运行时，前端通过 API 接口向后端发送请求，并接收后端返回的数据，实现了前后端数据的交互和协作。后端使用 MySQL 数据库存储数据，并提供 API 接口供前端调用，实现了数据的可持久化和可访问性。

### 1.3 监控视频摘要系统效果

监控视频摘要包括登录注册、智能分析、日志查询和视频图库共 4 个页面，系统的功能逻辑如图 2 所示。用户首先注册账号并登录，接着对目标区域进行监控。该系统支持用户上传监控视频进行分析。用户可以选择离线分析

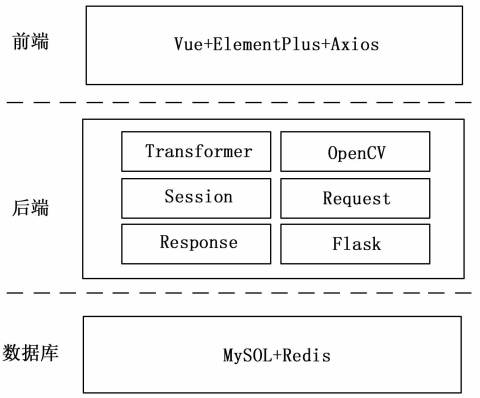


图 1　系统软件实现方案

模式，即上传监控视频，也可以选择连接摄像头实时检索目标行人。系统可以随时对监控的时间段进行智能分析，并保存日志，以供用户查询。此外，系统将监控的原视频和摘要视频保存至视频库，用户可以从视频库中指定某时间段或者全部时间，进行查询。下面分别对系统的登录注册、智能分析、日志查询和视频库页面进行展示。

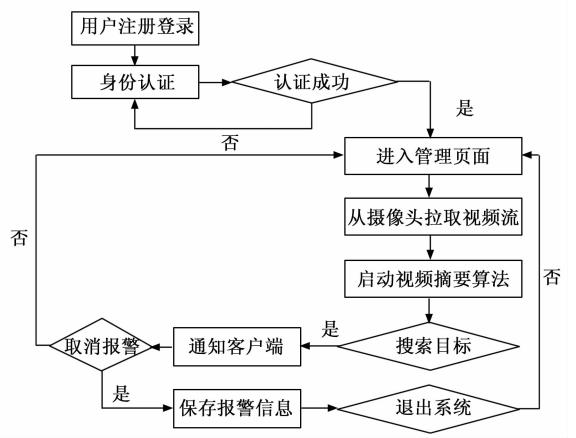


图 2　系统逻辑功能图

1.3.1　智能分析页面

用户登录成功后，即可进入视频摘要系统。监控视频摘要系统包括 3 个主页面：智能分析、日志查询和视频库，其中智能分析页面是监控视频摘要系统的核心页面，如图 3 所示，智能分析页面包括摘要生成、输出关键视频帧以及自动报警功能。下面对它们进行展示。

在智能分析页面，有两种模式，分别是离线模式和实时模式。其中离线模式可以上传客户想要分析得视频文件，点击选择文件就可以在本地上传自己的视频，然后点击上传就可以将视频信息输入到播放器进行准备播放，中间的视频摘要生成按钮，通过调用算法，将输入得视频进行摘要分析，在大约 2 s 后输出到右边的播放器里面，便于客户使用观看。

1.3.2　日志查询

日志查询页面用于保存用户的历史监控记录，图 4 展

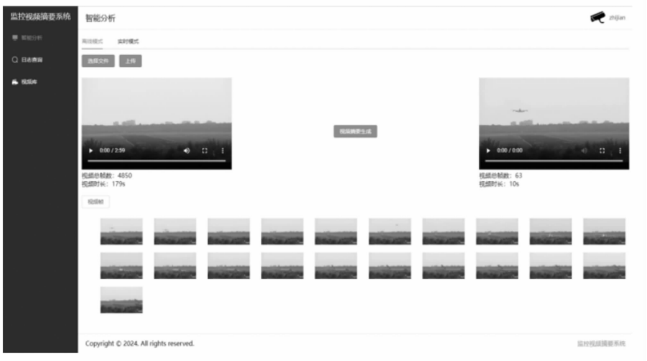


图 3　智能分析相关界面

示了系统日志查询的相关界面。用户可以根据监控场所名称和时间进行查询，点击详情按钮，会显示该监控的日志详细信息，即监控的视频摘要。也可以将过期的或者不需要的日志进行删除。


图 4　日志查询界面

1.3.3　视频库页面

视频库页面保存系统监控的所有视频，如图 5 所示，用户可以根据视频名称进行筛选。


图 5　视频库页面

## 2　Transformer 算法设计

### 2.1　基于动态 Transformer 的算法改进

Vision Transformers 将每个 2D 图像分割成 1D 标记，同时用自注意力机制建模它们的长距离交互。如前所述，为了正确识别一些图像并达到高精度，标记的数量通常需要很大，导致计算成本呈二次方增长。然而，构成大部分数据集的“更简单”图像通常需要更少的标记和更少的成本。受这一观察的启发，提出了动态 Transformer，旨在通过自适应地减少每个输入的代表令牌的数量来提高 Trans-

former 的计算效率。具体来说，部署多个使用越来越多的令牌进行训练的 Transformer，以便可以为每个测试图像依次激活它们，直到获得令人信服的预测（例如，具有足够的置信度）。计算在不同样本之间不均匀地分配，以提高整体效率。值得注意的是，如果单独学习所有 Transformer，一旦下游 Transformer 被激活，上游模型执行的计算就会被放弃，导致相当低的效率。为了缓解这个问题，引入了高效的特征和关系重用机制。动态 Transformer 模型设计如图 6 所示，动态 Transformer 是 Transformer 的一种变体，在视频摘要领域中具有重要的应用。相比于传统的 Transformer 模型，在处理视频序列时，动态 Transformer 能够减少计算量，提高模型的效率和性能。

相比于传统的 LSTM 网络，动态 Transformer 可以更好地处理长视频序列，同时减少计算量。在视频摘要领域中，动态 Transformer 被验证可以应用，帮助人们快速地获取视频摘要信息。

动态 Transformer 在视频摘要领域也有广泛的应用。与图像任务不同的是，视频摘要需要处理的是时间序列数据。在这种情况下，可以将多个具有不同时间粒度的动态 Transformer 层堆叠起来，如图 6 所示，以处理变长的视频序列。在测试时，它们被依次激活，直到获得令人信服的预测（如足够的信心）或推断出最终模型。一旦预测不能满足终止标准，原始输入视频的每一帧被分割成更多的时间段或片段，以便进行更准确但计算成本更高的推理。需要注意的是，在这里，每个时间片段的嵌入维度保持不变，而时间片段的数量增加，使其能够更细化地表示。

与图像任务类似，视频摘要中的动态 Transformer 也可以设计成具有相同结构但参数不同的多个层，以换取对一些 " 困难 " 测试样本的更高准确性，但这可能会牺牲计算成本。总之，动态 Transformer 在视频摘要领域中是一种非常有效的模型，可以处理变长的时间序列数据，并提高模型的效率和性能。

对于训练，只需训练动态 Transformer 在所有输出（即每个输出都有相应数量的 token）产生正确的预测。从形式上看，优化目标是：

$$\text{minimize}\ \frac{1}{|D_{\text{train}}|}\sum_{(x,y)\in D_{\text{train}}}\left[\sum_i L_{\text{CE}}(p_i,y)\right] \tag{1}$$

其中：$(x,y)$ 表示训练集 $D_{\text{train}}$ 中的一个样本及其相应的标签。采用标准的交叉熵损失函数 $L_{\text{CE}}(\cdot)$，而 $p_i$ 表示第 $i$ 个输出的 softmax 预测概率。

## 2.2 特征与关系重用

发展动态 Transformer 方法的一个重要挑战是如何促进计算的重复使用。也就是说，一旦推断出一个具有更多 token 的下游转化器，如果放弃在以前的模型中进行的计算，显然是低效的。上游模型虽然基于较少的输入 token，但都是以相同的目标进行训练的，并为完成任务提取了重要的信息。因此，提出了两种机制来重用学到的深度特征和自我注意力关系。这两种机制都能够通过最小的额外计算成本来大幅提高测试的准确性。

Transformer 编码器堆叠了 $L$ 个相同的块，每一个编码器主要由交替堆叠的多头注意力（MSA[12]，multi-head self-attention）和前馈神经网络（MLP[13]，multi-layer perceptron）组成。每一个模块都应用了残差连接[15]，后面紧跟层归一化（LN[14]，layer normalization）。$\boldsymbol{z}_l\in R^{N\times D}$ 表示第 $l$ 个 Transformer 层的输出，其中 $N$ 是每个样本的 token 数，$D$ 是每个 token 的尺寸。注意，$N=HW+1$，对应于原始图像的 $H\times W$ 个 token 和一个可学习的分类 token。从形式上看，有：

$$\begin{cases}\boldsymbol{z}'_l=\text{MSA}(\text{LN}(\boldsymbol{z}_{l-1}))+\boldsymbol{z}_{l-1}, l\in\{1,\cdots,L\}\\ \boldsymbol{z}_l=\text{MLP}(\text{LN}(\boldsymbol{z}'_l))+\boldsymbol{z}'_l, l\in\{1,\cdots,L\}\end{cases} \tag{2}$$

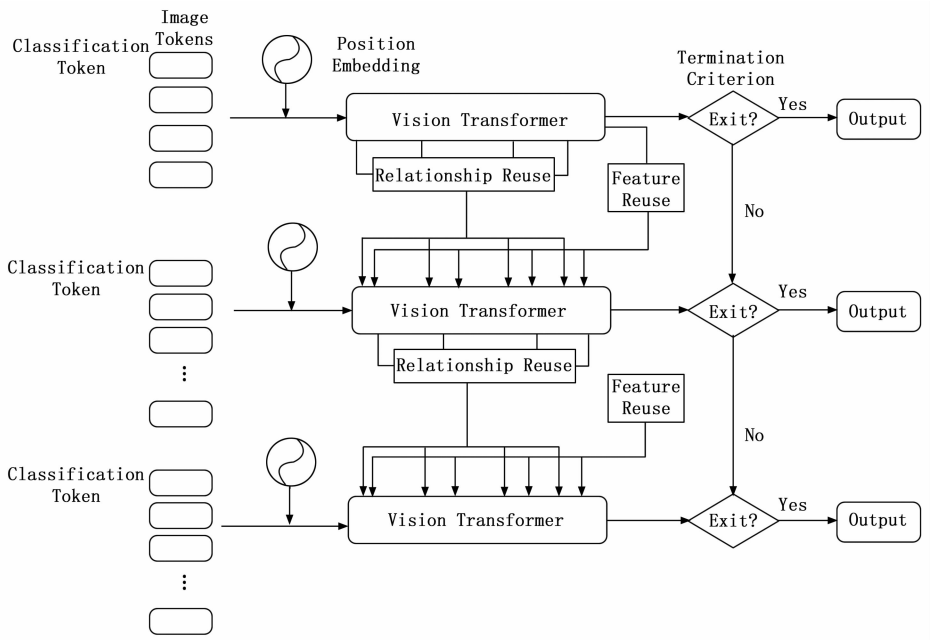


图 6 动态 Transformer 模型设计图

其中：$L$ 是 Transformer 中的总层数。$z_L$ 中的分类 token 将被送入 LN 层，然后再进入全连接层进行最终预测。为了简单起见，这里省略关于位置嵌入的细节，这与主要想法无关。除了配置之外，没有对它进行任何修改。

### 2.3　特征重用

动态 Transformer 中的所有 Transformer 都有一个共同的目标，即为准确识别而提取鉴别性的表征。因此，下游模型应该在之前获得的深度特征的基础上进行学习，而不是从头开始提取特征，这是非常直接的。前者更有效率，因为在上游模型中进行的计算对其本身和连续的模型都有帮助。为了实现这个想法，提出了一个特征重用机制如图 7 所示。具体来说，利用上游转化器的最后一层输出的视频帧 token，即 $z_L^{up}$，来学习下游模型的逐层嵌入 $E_l$。

$$E_l = f_l(z_L^{up}) \in R^{N\times D'} \tag{3}$$

在此，$f_l: R^{N\times D} \rightarrow R^{N\times D'}$，由一连串的操作组成，首先是 LN-MLP（$R^D \rightarrow R^{D'}$），它引入了非线性，允许更灵活地转换。然后，视频帧标记被重塑为原始视频帧中的相应位置，并进行上采样和扁平化，以匹配下游模型的 token 数量。通常情况下，使用一个小的 $D'$，以达到高效的 $f_l$。

因此，嵌入 $E_l$ 被注入下游模型，提供识别输入视频帧的先验知识。形式上，将公式（2）替换为：

$$\begin{cases} z_l = \mathrm{MLP}(\mathrm{LN}(\mathrm{Concat}(z'_l, E_l))) + z_l \\ l \in \{1, \cdots, L\} \end{cases} \tag{4}$$

其中：$E_l$ 与中间标记 $z'_l$ 相连接。只是将 LN 和 MLP 的第一层的维度从 $D$ 增加到 $D + D'$。由于 $E_l$ 是基于上游输出 $z_L^{up}$，它比 $z'_l$ 有更少的 token，它实际上为 $z'_l$ 中的每个 token 总结了输入视频帧的上下文信息。因此，将 $E_l$ 称为上下文嵌入。此外，不重复使用分类标记，并在公式（4）中对其进行置零，根据经验发现这对性能有好处。直观地说，公式（3）和（4）允许在最小化最终识别损失公式（1）的目标下，训练下游模型在每层基础上灵活地利用 $z_L^{up}$ 内的信息。这种特征重用的表述也可以被解释为隐含地扩大了模型的深度。

### 2.4　关系重用

Transformer 的一个突出优点是，其自我注意块能够整合整个图像的信息，这有效地模拟了数据中的长距离依赖关系。通常情况下，模型需要在每一层学习一组注意力图来描述 token 之间的关系。除了上面提到的深层特征外，下游模型还可以获得以前模型中产生的自我注意力图。认为这些学到的关系也能够被重用，以促进下游 Transformer 的学习。

考虑到输入表征 $z_l$，自我注意力是按以下方式进行的。首先，查询、键和值矩阵 $\boldsymbol{Q}_l$、$\boldsymbol{K}_l$ 和 $\boldsymbol{V}_l$ 是通过线性投影计算出来的。

$$\boldsymbol{Q}_l = z_l \boldsymbol{W}_l^Q, \boldsymbol{K}_l = z_l \boldsymbol{W}_l^K, \boldsymbol{V}_l = z_l \boldsymbol{W}_l^V \tag{5}$$

其中：$\boldsymbol{W}_l^Q$，$\boldsymbol{W}_l^K$ 和 $\boldsymbol{W}_l^V$ 是权重矩阵。然后，注意力图是通过一个带有 softmax 的缩放点乘运算来计算所有 token 的值，即：

$$\begin{cases} \mathrm{Attention}(z_l) = \mathrm{Softmax}(A_l)\boldsymbol{V}_l \\ A_l = \boldsymbol{Q}_l \boldsymbol{K}_l^T / \sqrt{d} \end{cases} \tag{6}$$

这里 $d$ 是 $\boldsymbol{Q}$ 或 $\boldsymbol{K}$ 的隐藏维度，$A_l \in \mathbb{R}^{N\times N}$ 表示注意力图的权重。注意，为了清楚起见，省略了关于多头注意机制的细节，其中 $A_l$ 可能包括多个注意图。这样的简化并不影响对方法的描述。

对于关系重用，首先将上游模型的所有层产生的注意力权重（即 $A_l^{up}, l \in \{1, \cdots, L\}$）连接起来。

$$A^{up} = \mathrm{Concat}(A_1^{up}, A_2^{up}, \cdots, A_L^{up}), \in R^{N_{up}\times N_{up}\times N_{up}^{Au}} \tag{7}$$

其中：$N_{up}$ 和 $N_{up}^{Au}$ 分别表示上游模型中的 token 和所有

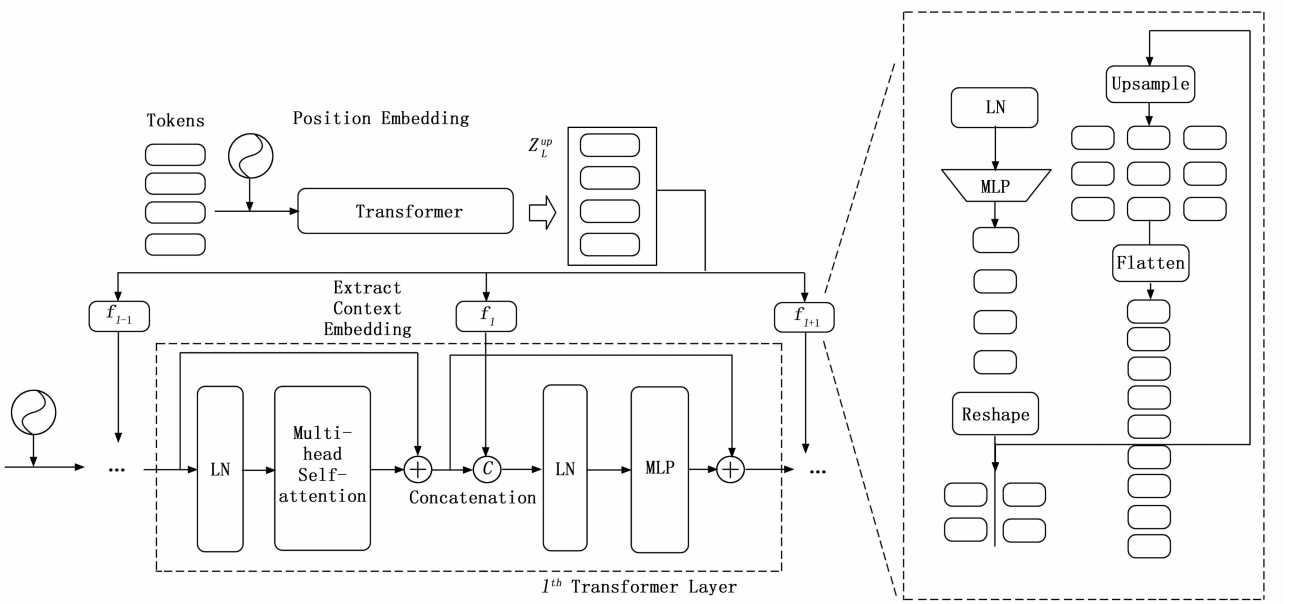


图 7　特征重用模型图

注意力图的数量。通常，有 $N_{up}^{Att}=N^{H}L$，其中 $N^{H}$ 是多头注意的头数，$L$ 是层数。然后，下游 Transformer 通过利用自己的 token 和 $A^{up}$ 来学习注意力图。形式上代替 $E_q$。

$$\begin{cases}\text{Attention}(\boldsymbol{z}_l)=\text{Softmax}(A_l+r_l(A^{up}))V_l\\ A_l=\boldsymbol{Q}_l\boldsymbol{K}_l^T/\sqrt{d}\end{cases}\tag{8}$$

其中：$r_l(\cdot)$ 是一个转换网络，它整合了 $A^{up}$ 提供的信息，以完善下游的注意力权重 $A_l$。$r_l(\cdot)$ 的结构包括一个用于非线性的 MLP，然后是一个上采样操作以匹配注意力图的大小。对于多头注意力，MLP 的输出维度将被设定为头的数量。

## 3 实验设置

### 3.1 数据集

本文使用了两个视频摘要的公共基准：SumMe[16] 和 TVSum[17] 来评价本文提出的方法。SumMe 由 25 个视频组成，覆盖了各种事件，如假期和烹饪。SumMe 中的视频长度从 1.5～6.5 min 不等。SumMe 中的每个视频由 15～18 人注释，并以关键镜头作为摘要。TVSum 有来自 TRECVID 多媒体事件检测（MED）的 10 个类别的 50 个视频（例如，养蜂和游行）。TVSum 中的视频长度从 2～10 分钟不等。类似地，每个视频由 20 个人用帧级重要性评分进行注释。此外，YouTube 和 OVP 用于训练。YouTube 上的视频涵盖体育、新闻等多种主题，而 OVP 上的视频多为纪录片。YouTube 有 39 个视频（不包括漫画），OVP 有 50 个视频，这些视频都用基于关键帧的摘要进行了注释，具体情况见表 3，关键帧被转换成重要性分数用于训练。在先前关于视频摘要的工作基础上，将提出的方法与现有的 3 种设置进行了比较：规范设置、扩展设置和转移设置。具体地说，规范设置是标准的监督学习设置，其中训练集和测试集是一个数据集的子集。考虑到 SumMe 和 TVSum 的训练量相对较小，采用了扩展的设置来增加训练样本。在此设置中，YouTube 数据集和 OVP 数据集中的视频被添加到训练集中，测试集保持不变。在转移设置方面，利用该方法来度量总结模型的转移能力，并在 SumMe（TVSum）上对模型进行训练，在 TVSum（SumMe）上对模型进行测试。注意，YouTube 和 OVP 两个数据集中的视频也增加了训练样本。

表 3 数据集汇总表

| DataSet | Number of videos | Duration/min | Topic | Annotation |
|---|---|---|---|---|
| SumMe | 25 | 1～6 | 假期、烹饪 | 15 组关键镜头 |
| TVSum | 50 | 2～10 | 游行、养蜂 | 20 组重要得分 |
| YouTube | 50 | 1～10 | 体育、新闻 | 5 组关键帧 |
| OVP | 39 | 1～4 | 纪录片 | 5 组关键帧 |

### 3.2 实验细节

根据文献［2］统一将视频每秒钟显示 2 帧图像，并利用 GoogleNet[18] 中的池化层将视频中的每一帧图像进行特征提取，特征维度为 1 024。对于 SumMe 数据集，真实注释以关键镜头的形式提供，因此直接使用这些真实摘要进行评估。但是，TVSum 数据集中缺少关键镜头注释。TVSum 提供由多个用户注释的帧级重要性分数。为了将重要性分数转换为基于关键镜头的摘要，遵循文献［2］中的过程，其中包括以下步骤：1）使用 KTS[2] 对视频进行时间分割以生成不相交的间隔；2）计算平均间隔得分并将其分配给间隔中的每个帧；3）根据分数对视频中的帧进行排名；4）应用背包算法来选择帧，使总长度低于一定的阈值，从而得到该视频的基于关键镜头的视频摘要。使用这种基于关键镜头的注释，通过选择具有最高重要性分数的帧来获取用于训练的关键帧。请注意，基于关键帧和基于关键镜头的摘要均表示为长度等于视频中帧数的 0/1 向量。这里，标签 0/1 表示是否在摘要视频中选择了帧。

测试时，将 $T=320$ 帧的均匀采样测试视频输入到模型，以获得长度为 320 的输出。然后，使用最近邻方法将该输出缩放为视频的原始长度

### 3.3 评价指标

按照文献［1］中的方法，本文使用基于镜头的评价指标。假设 $S_O$ 是生成的摘要，$S_G$ 是真实摘要。使用时间上的重叠来计算精度（$P$）和召回率（$R$），最后使用分数 $F=(2P\times R)/(P+R)$ 来作为评价指标。$P$ 和 $R$ 的表达式为：

$$\begin{cases}P=\dfrac{S_O\cap S_G}{S_O}\\ R=\dfrac{S_O\cap S_G}{S_G}\end{cases}\tag{9}$$

最后，随机抽取 20% 的数据用于测试，其余 80% 的数据用于训练和验证。由于数据是随机拆分的，因此在多个随机拆分的数据中重复实验，并得到平均 $F$ 分数性能。

本文为了衡量模型的计算复杂度，通过计算模型中所有层的权重和偏置值的乘积之和来计算。这个值就代表了模型的总 FLOPs 数。thop 库支持计算各种类型的模型结构，包括卷积层、全连接层、池化层等。它也可以处理包含多个输入和多个输出的复杂模型结构。因此，使用 thop 库可以方便地获取模型的计算量信息，帮助进行模型优化和性能评估。

### 3.4 实验结果与分析

规范设置下，表 4 显示了规范设置下 SumMe 和 TVSum 数据集上的 $F$ 分数（%）和参数（百万）与最先进的视频摘要方法的比较。对比方法可分为两类：一类是常规方法（表格上半部分），包括 Video MMR、LiveLight、ERSUM、MSDS-CC；另一种是一些深度学习方法，包括 dpplSTM 、vs-LSTM、SUM-GAN$_{dpp}$、A-AVS 、SUMGAN$_{sup}$、M-AVS、SASUM、SASUM$_{sup}$、DRDSN、DRDSN$_{sup}$、FCSN 、TS-STN 、VASNet、DSNet。清楚地观察到，本文的方法产生了最佳性能，在其中 SumMe 数据集中，$F$ 分数从 50.3% 提高到 53.9%，提高了 3.6%，在 TvSum 数据集中，从 62.1% 提高到 63.0%，提高了 0.9%。同时，本文的模型大小只有 1.53 M，小于对比方法中最小

模型的 2.63 M。这表明所提出的模型在效率和性能方面具有较大的优势。表 5 显示了规范、扩展、转移设置下 TVSum 数据集上 $F$ 分数（%）与 DSNet 等的比较。为了得到更多的性能验证结果，OVP 和 YouTube 数据集上重现了 DSNet，并与动态 Transformer 进行了比较。在 5 次试验的 OVP 数据集上，动态 Transformer 实现了较高的平均 $F$ 分数（动态 Transformer 为 41.7%，DSNet 为 34.3%）和较低标准差（动态 Transformer 为 0.043，DSNet 为 0.062）。尽管在 YouTube 数据集上动态 Transformer 的标准差略高于 DSNet（动态 Transformer 为 0.051，DSNet 为 0.038），但 5 个试验中的平均 $F$ 分数（动态 Transformer 为 37.2%，DSNet 为 32.4%）在平均 $F$ 分数明显高于 DSNet。

表 4　规范设置下 SumMe 和 TVSum 数据集上最先进的视频摘要方法的 $F$ 分数（%）和参数（百万）比较

| Method | $F$ on SumMe | $F$ on TvSum | Params/M |
|---|---|---|---|
| Video MMR | 26.6 | | |
| LiveLight | | 46.0 | |
| ERSUM | 43.1 | 59.4 | |
| MSDS-CC | 40.6 | 52.3 | |
| vsLSTM | 37.6 | 54.2 | 2.63 |
| dppLSTM | 38.6 | 54.7 | 2.63 |
| SUM-GANdpp | 39.1 | 51.7 | 295.86 |
| SUM-GANsup | 41.7 | 56.3 | 295.86 |
| A-AVS | 43.9 | 59.4 | 4.40 |
| M-AVS | 44.4 | 61.0 | 4.40 |
| SASUM | 40.6 | 53.9 | 44.07 |
| SASUMsup | 45.3 | 58.2 | 44.07 |
| DR-DSN | 41.4 | 57.6 | 2.63 |
| DR-DSNsup | 42.1 | 58.1 | 2.63 |
| TS-STN | 46.1 | 60.0 | 16.18 |
| FCSN | 48.8 | 58.4 | 152.07 |
| VASNet | 49.7 | 61.4 | 7.35 |
| DSNet | 50.2 | 61.2 | 8.53 |
| Ours | 53.9 | 63.0 | 1.53 |

表 5　在规范、扩展和转移设置下 TVSum 数据集上的 $F$ 分数（%）与最先进的视频摘要方法的比较

| Method | Canonical | Augmented | Transfer |
|---|---|---|---|
| vsLSTM | 54.2 | 57.9 | 56.9 |
| dppLSTM | 54.7 | 59.6 | 58.7 |
| SUM-GANsup | 56.3 | 61.2 | |
| A-AVS | 59.4 | 60.8 | |
| M-AVS | 61.0 | 61.8 | |
| FCSN | 58.4 | 59.1 | 57.4 |
| DSNet | 62.1 | 63.9 | 59.4 |
| Ours | 63.0 | 64.0 | 59.4 |

### 3.5　消融实验

采用有监督的方式对本文提出的模块进行了消融实验，以验证其有效性，如表 6 所示，特征重用的思想是将上游 Transformer 最后一层输出的特征取出，经 MLP 变换和上采样（Upsample）后，作为上下文嵌入（Context Embedding）以 Concat 的方式整合入下游模型每一层的 MLP 模块中。利用上游 Transformer 最后一层中输出的 token（$z_{up}$）来学习下游模型中的 layer-wise Embedding（El）。故在这一部分去除后，当上游 Transformer 的输出不够精确时候，需要重新学习不同的特征。

表 6　去除特征重用数据集比较　　%

| Module | $F$ on SumMe | $F$ on TVSum | FLOPs/G |
|---|---|---|---|
| Ours-feature reuse | 41.4 | 57.6 | 1.78 |
| Ours | 63.0 | 64.0 | 1.1 |

从表 6 中可以看出，去掉特征重用后，要重新计算不同的 Transformer 中的特征，这样显然效率很低，而且复杂度上升，从 1.1 G 上升到 1.78 G。上游模型虽然基于较少数量的输入标记，但以相同的目标进行训练，并提取了完成任务的有价值的信息。

本文再对注意力重用进行消融实验，以验证其有效性，如表 7 所示，注意力重用的核心思想是将上游模型的全部注意力图以 logits 的形式进行整合，经 MLP 变换和上采样后，加入下层每个注意力图的 logits 中。这样，下游模型每一层的注意力模块都可灵活复用上游模型不同深度的全部注意力信息，且这一复用信息的强度可以通过改变 MLP 的参数自动地调整。值得注意的是，对注意力图进行上采样需要对其行或列进行重组后分别完成，以确保其几何关系的对应性。

表 7　去除注意力重用数据集比较

| Module | $F$ on SumMe | $F$ on TVSum | FLOPs/G |
|---|---|---|---|
| Ours-attention reuse | 41.4 | 57.6 | 1.83 |
| Ours | 63.0 | 64.0 | 1.1 |

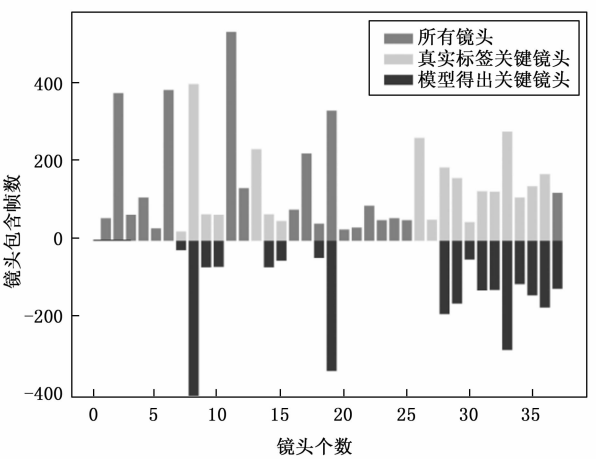


图 8　模型在 TVSum 数据集上的视频摘要可视化

从表 3～5 中可知，取消注意力重用后，模型的复杂度明显上升。因为由于注意力权重矩阵需要对输入序列进行

计算，因此在每个位置上都需要进行类似的计算，这会使得模型的计算量和存储量变得非常大。

为了验证提出的模型在监控领域具有应用价值，找来一段监控视频如图 9 所示，利用模型进行视频摘要进行提取，左上角为原始视频的第 65 帧，左下角是原始视频第 130 帧，右下角为原始视频的第 191 帧，右上角为大纲视频的第 65 帧其中右边的运动对象与大纲视频相同。从图中可以看到上，大纲视频第 65 帧有两个运动目标，左边的运动目标出现在原始视频的第 130 帧，右边的运动目标出现在原始视频的第 191 帧，可以很清晰地得出模型满足以满足精度要求和监控要求。

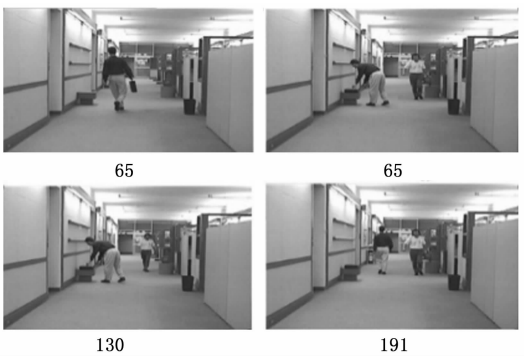


图 9 监控原始视频和浓缩视频对比示意图

## 4 结束语

本研究提出了一种基于动态 Transformer 的视频摘要模型，旨在解决传统监控方法中存在的梯度消失、计算复杂度高等问题。通过采用特征重用和注意力重用技术，本文有效降低了视频摘要算法的计算复杂度，同时提高了模型在长序列建模方面的性能。实验结果表明，相较于传统 Transformer 模型，本文的模型在准确率方面取得了显著的提升。此外，引入了特征重用和注意力重用，有效减少了冗余计算，使得模型更具实用性。

未来将考虑进一步优化模型性能，探索更有效的特征提取方法以及更复杂的视频内容和场景。此外，也将在不同数据集上进行更广泛的验证，以确保模型的泛化性能。

**参考文献：**

[1] JAIN A K, MURTY M N, FLYNN P J. Data clustering: a review [J]. ACM Computing Surveys (CSUR), 1999, 31 (3): 264 - 323.

[2] DE AVILA S E F, LOPES A P B, DA LUZ JR A, et al. VSUMM: A mechanism designed to produce static video summaries and a novel evaluation method [J]. Pattern Recognition Letters, 2011, 32 (1): 56 - 68.

[3] LIU H, LIU Y, YU Y, et al. Diversified key-frame selection using structured $L_{2,1}$ optimization [J]. IEEE Transactions on Industrial Informatics, 2014, 10 (3): 1736 - 1745.

[4] APOSTOLIDIS E, ADAMANTIDOU E, METSAI A I, et al. Video summarization using deep neural networks: a survey [J]. Proceedings of the IEEE, 2021, 109 (11): 1838 - 1863.

[5] EMAD A, BASSEL F, REFAAT M, et al. Automatic video summarization with timestamps using natural language processing text fusion [C] //Proceedings of Computing and Communication Workshop and Conference. Las Vegas, NV, USA: IEEE Press, 2021: 60 - 66.

[6] NARASIMHAN M, ROHRBACH A, DARRELL T. Clip-it! language-guided video summarization [J]. Advances in Neural Information Processing Systems, 2021, 34: 13988 - 14000.

[7] LE N, RATHOUR V S, YAMAZAKI K, et al. Deep reinforcement learning in computer vision: a comprehensive survey [J]. Artificial Intelligence Review, 2022: 1 - 87.

[8] JAMBLI M N, KHAN A S, SHOON S C. A survey of VASNET framework to provide infrastructure-less green IoTs communications for data dissemination in search and rescue operations [J]. Journal of Electronic Science and Technology, 2016, 14 (3): 220 - 228.

[9] HESAMIAN M H, JIA W, HE X, et al. Deep learning techniques for medical image segmentation: achievements and challenges [J]. Journal of Digital Imaging, 2019, 32: 582 - 596.

[10] PATIL K, BRAZDIL P. Sumgraph: text summarization using centrality in the pathfinder network [J]. International Journal on Computer Science and Information Systems, 2007, 2 (1): 18 - 32.

[11] APOSTOLIDIS E, ADAMANTIDOU E, METSAI A I, et al. Video summarization using deep neural networks: A survey [J]. Proceedings of the IEEE, 2021, 109 (11): 1838 - 1863.

[12] LIMA L R, GODEIRO L L. Equity-premium prediction: Attention is all you need [J]. Journal of Applied Econometrics, 2023, 38 (1): 105 - 122.

[13] ALAYRAC J B, DONAHUE J, LUC P, et al. Flamingo: a visual language model for few-shot learning [J]. Advances in Neural Information Processing Systems, 2022, 35: 23716 - 23736.

[14] SAHARIA C, CHAN W, SAXENA S, et al. Photorealistic text-to-image diffusion models with deep language understanding [J]. Advances in Neural Information Processing Systems, 2022, 35: 36479 - 36494.

[15] GUO M H, XU T X, LIU J J, et al. Attention mechanisms in computer vision: a survey [J]. Computational Visual Media, 2022, 8 (3): 331 - 368.

[16] LE N, RATHOUR V S, YAMAZAKI K, et al. Deep reinforcement learning in computer vision: a comprehensive survey [J]. Artificial Intelligence Review, 2022: 1 - 87.

[17] SONG Y, VALLMITJANA J, STENT A, et al. Tvsum: Summarizing web videos using titles [C] //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015: 5179 - 5187.

[18] GUO M H, XU T X, LIU J J, et al. Attention mechanisms in computer vision: a survey [J]. Computational Visual Media, 2022, 8 (3): 331 - 368.