Journal of Image and Signal Processing 图像与信号处理, 2024, 13(2), 190-209
Published Online April 2024 in Hans. https://www.hanspub.org/journal/jisp
https://doi.org/10.12677/jisp.2024.132017

Hans汉斯

基于时空Transformer的端到端的视频注视目标检测

彭梦昊，王　冠，徐　浩，景圣恩

合肥工业大学计算机与信息学院，安徽　合肥

收稿日期：2024年3月19日；录用日期：2024年4月9日；发布日期：2024年4月18日

摘　要

注视目标检测旨在定位人的注视目标。HGTTR的提出，将Transformer结构用于注视目标检测的任务中，解决了卷积神经网络需要额外的头部探测器的问题，实现了端到端的对头部位置和注视目标的同时检测，并且实现了优于传统的卷积神经网络的性能。然而，目前的方法在视频数据集上的性能还有较大提升空间。原因在于，当前的方法侧重于在单个视频帧中学习人的注视目标，没有对视频中的时间变化进行建模，所以无法解决动态注视、镜头失焦、运动模糊等问题。当一个人的注视目标在不断的发生变化时，缺乏时间变化建模可能会导致定位注视目标偏离人的真实注视目标。并且由于缺乏对于时间维度上的建模，模型无法解决因为镜头失焦和运动模糊等问题所导致的特征缺失。在这项工作当中，我们提出了一种基于时空Transformer的端到端的视频注视目标检测模型。首先，我们提出帧间局部可变形注意力机制，用于处理特征缺失的问题。其次，我们在可变形注意力机制的基础上，提出帧间可变形注意力机制，利用相邻视频帧的时序差异，动态选择采样点，从而实现对于动态注视的建模。最后，我们提出了时序Transformer来聚合由当前帧和参考帧的注视关系查询向量和注视关系特征。我们的时序Transformer包含三个部分：用于编码多帧空间信息的时序注视关系特征编码器，用于融合注视关系查询的时序注视关系查询编码器以及用于获取当前帧检测结果的时序注视关系解码器。通过对于单个帧空间、相邻帧间以及帧序列三个维度的时空建模，很好的解决了视频数据中常见的动态注视、镜头失焦、运动模糊等问题。大量实验证明，我们的方法在VideoAttentionTarget和VideoCoAtt两个数据集上均取得了较为优异的性能。

关键词

注视目标检测，Transformer，可变形注意力，时序变化建模

End-to-End Video Gaze Target Detection with Spatial-Temporal Transformers

Menghao Peng, Guan Wang, Hao Xu, Sheng'en Jing

School of Computer Science and Information Engineering, Hefei University of Technology, Hefei Anhui

文章引用: 彭梦昊, 王冠, 徐浩, 景圣恩. 基于时空 Transformer 的端到端的视频注视目标检测[J]. 图像与信号处理, 2024, 13(2): 190-209. DOI: 10.12677/jisp.2024.132017

Received: Mar. 19th, 2024; accepted: Apr. 9th, 2024; published: Apr. 18th, 2024

Abstract

Gaze target detection is designed to locate the human gaze target. Proposed by HGTTR, Transformer structure is used in the task of gaze target detection, which solves the problem that convolutional neural networks need additional head detectors, realizes the end-to-end simultaneous detection of head position and gaze target, and achieves better performance than traditional convolutional neural networks. However, there is still much room for improvement in the performance of current methods on video data sets. The reason is that the current method focuses on learning the human gaze target in a single video frame, and does not model the time change in the video, so it cannot solve the problems of dynamic gaze, out-of-focus lens, and motion blur. When a person's gaze target is constantly changing, the lack of time change modeling may cause the fixed gaze target to deviate from the person's real gaze target. In addition, due to the lack of modeling in the time dimension, the model cannot solve the feature loss caused by out-of-focus lens and motion blur. In this work, we propose an end-to-end video gaze target detection model based on spatial-temporal Transformers. First, we propose an interframe local deformable attention mechanism to deal with feature missing problems. Secondly, on the basis of the deformable attention mechanism, we propose the Inter-frames deformable attention mechanism, which uses the timing difference of adjacent video frames to dynamically select sampling points, so as to realize the modeling of dynamic gaze. Finally, we propose a temporal Transformers to aggregate gaze relation query vectors and gaze relation features from the current frame and reference frame. Our temporal Transformers consists of three parts: A temporal gaze feature encoder for encoding multi-frame spatial information, a temporal gaze query encoder for fusing gaze queries, and a temporal gaze decoder for obtaining current frame detection results. Through the spatial-temporal modeling of single frame space, adjacent frames and frame sequence, the common problems of dynamic gaze, lens out of focus and motion blur in video data are solved well. A large number of experiments show that our method achieves excellent performances on both VideoAttentionTarget and VideoCoAtt datasets.

Keywords

Gaze Target Detection, Transformer, Deformable Attention, Temporal Variation Modeling

1. 引言

注视目标检测是计算机视觉领域中的一项重要的研究任务，旨在定位图像(视频帧)中的每个人物的头部位置和相应的注视目标位置，是很多视觉任务的基础性工作。视频注视目标检测旨在视频数据中检测每个视频帧中所有人物的注视目标。随着深度学习技术的发展和计算资源的不断提升，视频注视目标检测在自动驾驶、智能监控、人机交互等领域具有广泛的应用前景，因而近年来引起了人们较大的研究兴趣。

基于深度学习的注视目标检测方法能够克服基于几何特征的方法[1]对于使用环境存在限制的问题，因为后者必须通过眼部特征对视线进行估计。利用深度网络对场景和头部姿态等信息进行提取和处理，

实现了不需要佩戴视线追踪设备、在非限制性场景下对注视目标进行检测。在这个方向上的关键一步是Recasens 等人[2]的工作，这项工作提出了第一个相关数据集 GazeFollow，并提出了一个双分支卷积神经网络，其第一个分支用于估计图像场景显著性特征，第二个分支用于获取头部位置及头部姿态特征。在随后的几项工作中，[3] [4] [5] [6] [7]均采用了这种双分支结构的设计方式，并进一步引入了额外的组件。例如，Chong 等人[3]在双分支结构的基础上引入了检测场景中不存在注视目标的情况(即注视目标在帧外)。随后，同样的作者[5]将 CNN-LSTM 集成到双分支结构的模型中，对视频中的注视关系进行动态建模，并提出了 VideoAttentionTarget 数据集。与[5]方法相似的是 Fan 等人[8]的方法，不同的是，他们研究的是在社交场景中的共同注视问题(两人或两人以上注视同一目标)，并提出 VideoCoAtt 数据集。

一些研究除了从场景显著性和头部姿态中获得外，还加入了从单目深度估计器中获得的深度图[4] [9] [10] [11]，例如 Fang 等人[12]整合了头部姿态、眼部姿态及场景深度信息。Jin 等人[13]，在提取场景和头部特征的基础上利用场景图深度信息和三维视线方向的基础上，利用残差特征过滤模块进一步增强各特征之间的信息融合。

与上述方法不同，Tu 等人[14]提出的基于 Transformer 的注视目标检测方法，其使用卷积神经网络提取图像的高级特征，将图像特征送入 Transformer 编码器中强化空间特征，并利用解码器并行解码个注视关系查询构成的查询序列，取得了优于基于卷积神经网络方法的所有方法。Tonini 等人[15]，在 Tu 等人[14]的基础上，通过建立图像每个人物和被注视目标之间的关系，引入目标感知注意力机制，在公开数据集上取得了最好的结果。相同的作者[16]还研究了注视目标检测中的跨域问题。

但是我们可以看出，现有的方法更多的是处理静态图像中的信息，很少关注在视频中动态的注视关系问题，即在连续的视频帧中，人物的头部姿态变化较小，但注视位置变化较大的问题。尽管 Chong 等人[4]的方法，通过 Conv-LSTM 对视频中的注视关系进行动态建模，但其采用的两分支结构需要一个额外的头部探测器，会大大降低在现实的应用性。[14] [15]提出的方法只利用 Transformer 在空间范围内进行建模，由于缺乏对时序信息的建模，这些方法不能很好地处理在视频中出现的注视转移、运动模糊、部分遮挡等情景。这在 Tonini 等人[15]的实验结果中有较为明显的体现，其提出的方法在图片数据集(GazeFollow)中有非常明显的性能提升，但在视频数据(VideoAttentionTarget)中的性能提升较小。所以我们认为对时序信息进行建模，对于视频中的注视目标检测是十分有必要的。

为了更好地解决在视频中的注视目标检测问题，我们提出了基于时空 Transformer 的端到端的视频注视目标检测模型，目的是采用端到端的方式检测视频序列中每一个视频帧中所有人的注视目标位置，我们的模型利用 Transformer 从空间和时间两个维度进行建模，将视频注视目标检测视为端到端的序列解码问题。根据 Tu 等人[14]的设计，我们将输出表示为个注视关系实例，格式为<头部位置，注视目标位置>，其中为当前帧中的人数。所以视频注视目标检测可以表述为一个基于集合的预测问题，我们定义了一个具有多个可学习嵌入的注视关系查询序列，作为解码器的输入，每个注视关系查询被设计为最多捕获一个注视关系实例。

为了更好的建模时序信息，受到 Long 等人[17]设计的 SIFA 块和 Deformable DETR [18]的启发，设计了帧间可变形注意力机制和帧间局部可变形注意力机制，两种机制都利用相邻两帧之间的时序差异，前者对动态的注视关系进行更好的建模，后者对当前帧中每个空间位置的特征进行加强。除此之外，我们还设计了时序可变形注意力机制有效地聚合多帧空间信息。

从模型整体上看，我们的模型包含特征提取模块、帧间可变形 Transformer、时序 Transformer 和一个预测模块。特征提取模块利用卷积神经网络从单个视频帧中提取高级图像特征，之后利用帧间局部可变形注意力机制和帧间可变形注意力机制构建的帧间可变形 Transformer 对帧注视关系查询向量进行学习，然后通过时序 Transformer 将每个视频帧的注视关系查询向量和注视关系特征同时连接起来。时序

Transformer 包含三个部分：用于编码多帧空间信息的时序注视关系特征编码器，用于融合注视关系查询的时序注视关系查询编码器以及用于获取当前帧检测结果的时序注视关系解码器。这些模块为每个帧共享，并且可以端到端方式进行训练。通过这样的设计，可以很好地探索时间信息对动态注视关系的影响。总的来说，本文的贡献如下：

1) 我们提出了一种新的针对视频注视目标检测的模型，它可以以端到端的方式检测视频中每帧中所有人物的注视目标。

2) 我们设计了帧间可变形注意力机制和帧间局部可变形注意力机制，通过关注相邻帧的时序变化，对动态的注视关系以及模糊特征进行更好的建模，提高了模型的鲁棒性。

3) 我们设计了一种时序 Transformer，同时对多帧的空间特征和注视关系查询进行融合，实现帧序列层次的时间建模。

4) 我们的模型设计了基于相邻帧间和基于视频帧序列间两种融合时间信息的的方式，并在 VideoAttentionTarget 和 VideoCoAtt 两个数据集上取得了较好的结果。

2. 相关工作

2.1. 注视目标检测

本文的主要任务是从第三人称视角采集的视频中检测每个视频帧中所有人物的头部位置及其注视目标位置。对于注视目标检测任务，Recasens 等人[2]提出了第一个相关数据集 GazeFollow，并提出了一个融合场景显著性特征和头部姿态特征的双分支卷积神经网络，在随后的几项工作中，[3]-[8]均采用了这种双分支结构的设计方式，并进一步引入了额外的组件。Saran 等人[19]将这种方法应用于人机交互任务中，Chong 等人[3]将其扩展至检测注视目标不在帧内的情况。后来，同样的作者[5]将 CNN-LSTM 集成到双分支结构中，对视频中的动态注视关系进行建模，并提出 VideoAttentionTarget 数据集。与[5]方法相似的是 Fan 等人[8]的方法，不同的是，他们研究的是在社交场景中的共同注视问题(两人或两人以上注视同一目标)，并提出 VideoCoAtt 数据集。一些研究除了从场景显著性和头部姿态中获得外，还加入了从单目深度估计器中获得的深度图[4] [9] [10] [11]，我们可以称之为三分支结构，例如 Fang 等人[12]整合了头部姿态、眼睛位置及场景深度信息。Jin 等人[13]，在提取场景和头部特征的基础上利用场景图深度信息和三维视线方向的基础上，利用残差特征过滤模块进一步增强各特征之间的信息融合。与上述方法不同的是 Tu 等人[14]提出的基于 Transformer 的检测方法，其使用预选训练好的卷积神经网络作为特征提取主干网络，利用 Transformer 具有全局感受野的特点，更好地建模长距离的依赖关系，取得了优于基于卷积神经网络方法的所有方法的结果。Tonini 等人[15]，在此基础上，通过建立图像中每个人物和被注视目标直接的关系，引入目标感知注意力，在公开数据集上取得了最好的结果。但是，除 Chong 等人[5]的工作之外，现有的方法没有考虑时序信息对于视频中注视目标检测的影响，然而[5]中的方法仍采用双分支的结构，其不得不需要一个额外的头部检测器，无法实现端到端的检测。对于视频中的动态注视关系，对时序信息建模是有必要的。

2.2. Transformer

Transformer [20]最早是为了解决自然语言处理中的相关问题而提出的。以自注意力机制为关键组件，其具有捕获长期依赖的能力。所以利用 Transformer 全局建模的优点，将 Transformer 应用于计算机视觉任务，成为近年来的研究热点[21]。Dosovitskiy 等人[22]提出了一种纯 Transformer 架构(ViT)用于解决图像分类方面的问题，并取得了最优的性能。Carion 等人[23]利用 Transformer 实现了目标检测算法 DETR，其将目标检测问题视为一个集合预测的问题，设置指定长度的目标查询集合在解码器中与编码器的输出

做交叉注意力将图像特征序列转换成了包含对象级别信息的集合序列，最后利用多层感知机层从目标查询集合产生预测结果。Zhu 等人[18]设计 Deformable DETR，通过多尺度可变行注意力机制用于解决 DETR 对于小目标不敏感、计算量较大以及训练周期较长的问题，并取得了优异的性能。

对于注视关系的研究，Cheng [24]等人首次将 Transformer 结构应用于视线估计任务中，其使用 Transformer 的编码器结构处理人的头部图像，在视线估计任务中实现了最优性能。Tu 等人[14]首次将 Transformer 应用于注视目标检测的任务中，其沿用 DETR 的结构和预测模块设计，用“人物-注视目标”查询替代目标查询，能够同时检测人类头部的位置及其注视目标的位置，采用端到端的方式实现了注视目标检测任务。但是，目前的方法都仅利用 Transformer 结构对空间上下文进行建模，忽略了视频中的时序信息对于注视目标检测的影响。

3. 端到端视频注视目标检测任务设定

根据引言的描述可知，视频注视目标检测的任务是检测出每个视频帧中所有的人物头部位置和人物的注视目标位置，因为我们的目标是同时检测人物的位置和人物的注视目标的位置，所以我们将两者构成一个检测对<人物头部位置，注视目标位置>，并命名为“注视关系实例”，所以我们的任务转化为检测出视频帧中所有的“注视关系实例”。

根据上述对于任务的转化以及 Tu 等人[14]的设计，我们将视频注视目标检测问题定义为一个基于集合的预测问题。具体来说，给定一段视频，视频中的每个视频帧包含一个或多个人物以及相应的场景，我们将当前视频帧和其前后几个连续的参考帧作为输入，我们的模型被设计为检测出当前视频帧的所有人物的头部位置以及他们的注视目标，即检测出所有的“注视关系实例”。其输出可以用 N 个注视关系实例来表示，其中 N 为当前视频帧中人物的数量。在数学上，我们通过下面四个向量来定义每个注视关系实例：1) 一个由相应的视频帧大小归一化的头部边界框向量 $l_h \in [0,1]^4$；2) 一个表示是否头部的概率值 $p_h \in [0,1]$ 3) 一个表示帧外注视(注视目标在帧外)的概率值 $p_g \in [0,1]$；4) 一个注视目标热图向量 $l_g \in [0,1]^{H_o \times W_o}$，$H_o$ 和 W_o 表示输出注视目标热图的空间分辨率。

具体来说，对于当前视频帧 $x_{t_0} \in \mathbb{R}^{3\times H\times W}$，$t_0$ 为当前视频帧的时间索引，我们将其前后各 i 个视频帧作为其参考帧，将视频帧 x_{t_0} 和 $2i$ 个参考帧构成一个连续的视频帧序列 X。对于视频帧序列中的每个视频帧，我们将其表示为 $x_t \in \mathbb{R}^{3\times H\times W}$，$t \in T$，$T = \{t_0 - i, \cdots, t_0 - 1, t_0, t_0 + 1, \cdots, t_0 + i\}$。我们同时检测当前视频帧 x_{t_0} 中所有人的头部位置 l_h，及他们的注视目标位置 l_g。头部位置 l_h 被表示为一个边界框 $(x_{lt}, y_{lt}, x_{rb}, y_{rb})$，其中 x 和 y 分别表示横纵坐标，下标 lt 和 rb 分别表示边界框的左上角和右下角，即用左上角二维坐标和右下角二维坐标确定唯一的边界框。同时，注视目标 l_g 用高斯热图表示。

4. 方法

整体结构如图 1 所示，模型以当前帧和周围几个参考帧构成一个视频帧序列作为输入，输出当前帧(中间视频帧)中所有注视关系实例的检测结果。模型主要由以下四个部分构成：特征提取模块、帧间可变形 Transformer、时序 Transformer、预测模块。特征提取模块：由各视频帧共享参数的 CNN 主干网络用来提取视频帧的特征。帧间可变形 Transformer：以相邻两帧的特征图为输入，建模两帧中前一帧的注视关系特征以及注视关系查询向量。时序 Transformer 包含三个组成部分：时序注视关系特征编码器用来融合来自所有的帧间可变形编码器输出的注视关系特征，形成时序注视关系特征，作为时序注视关系查询的空间线索。时序注视关系查询编码器沿时间维度连接所有的空间解码器输出的注视关系查询向量。时序注视关系解码器以时序注视关系查询编码器和时序注视关系特征编码器的输出为输入，解码产生当前视频帧的最终的注视关系查询向量，用于预测模块产生最终的预测结果。

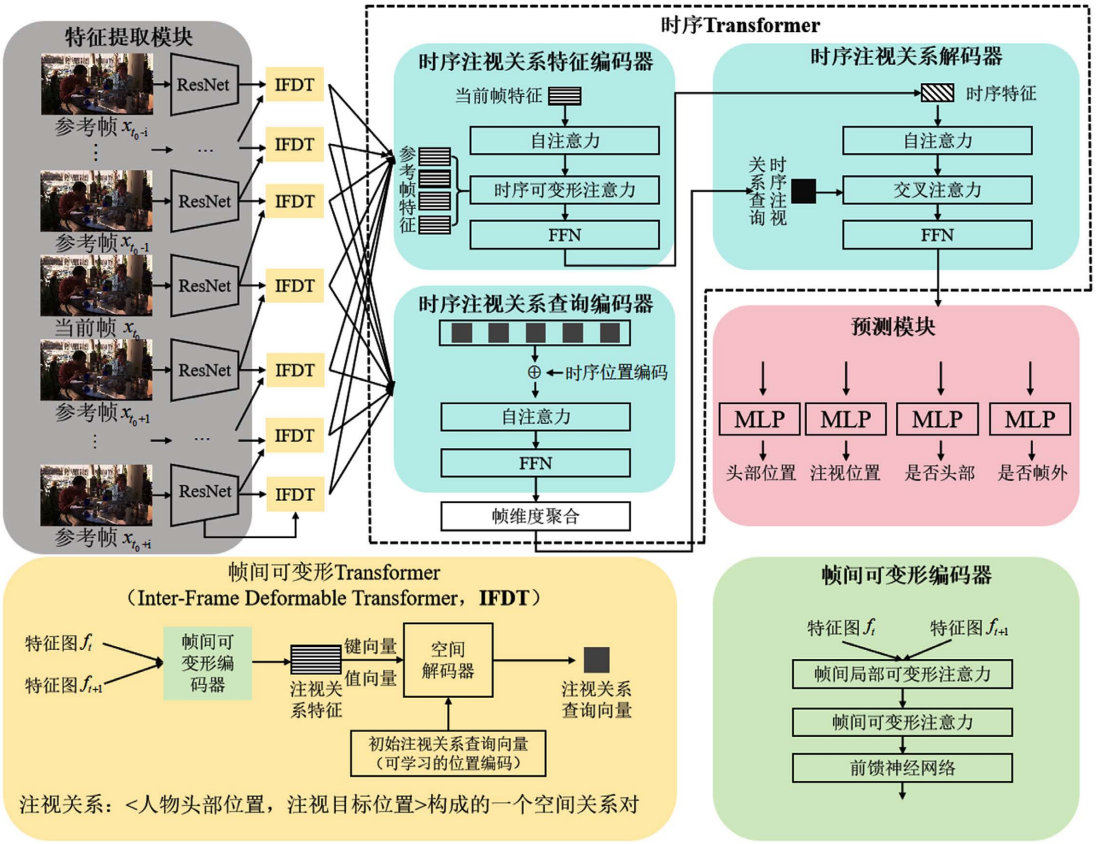

Figure 1. Overall structure of the model
图 1. 模型整体结构图

4.1. 特征提取模块

对于特征提取模块，我们使用 ResNet-50 [25]作为主干网络来提取视频帧的特征。具体来说，对于视频帧 $x_t \in \mathbb{R}^{3\times H\times W}$，通过 ResNet-50 得到特征图 $f_t \in \mathbb{R}^{C\times H'\times W'}$，然后将 f_t 送入一个卷积核大小为 1 × 1 的二维卷积层中，将通道维度由 C 降低到 C'，表示为 $f_t' \in \mathbb{R}^{C'\times H'\times W'}$。其中 $C=2048$，$C'=256$，t 代表时序索引，H 代表高度，W 代表宽度。

4.2. 帧间可变形 Transformer

根据 Tu 等人[14]的设计方式，我们的帧间可变形 Transformer 采用“编码器-解码器”的结构进行设计。具体来说包含一个帧间可变形编码器和一个空间解码器。Tu 等人在 HGTTR 中设计的编码器和解码器结构使用了标准的 Transformer 结构，核心是多头自注意力机制和多头交叉注意力机制，与之不同的是，我们把标准的 Transformer 编码器中的自注意力机制替换为我们设计的帧间可变形注意力机制，提高模型对于动态注视关系的学习能力。我们还设计了帧间局部可变形注意力机制，提高了模型对于模糊特征的学习能力。空间解码器使用标准的 Transformer 解码器结构。

4.2.1. 帧间局部可变形注意力机制

我们从 SIFA [17]中获得启发，提出帧间可变形注意力机制，我们的帧间局部可变形注意力机制是为了利用相邻帧的局部特征信息，对当前帧的模糊特征进行加强。我们通过融合当前查询元素和在相邻帧中以当前查询元素为中心的局部区域内的特征，对当前查询元素的特征进行加强。然而，简单地在相同

大小的局部区域($k \times k$ 网格)上使用注意力机制，容易忽略物体在不同帧中的不规则变换。为了缓解这一问题，我们设计了帧间局部可变形注意力模块，工作机制如图 2 所示。

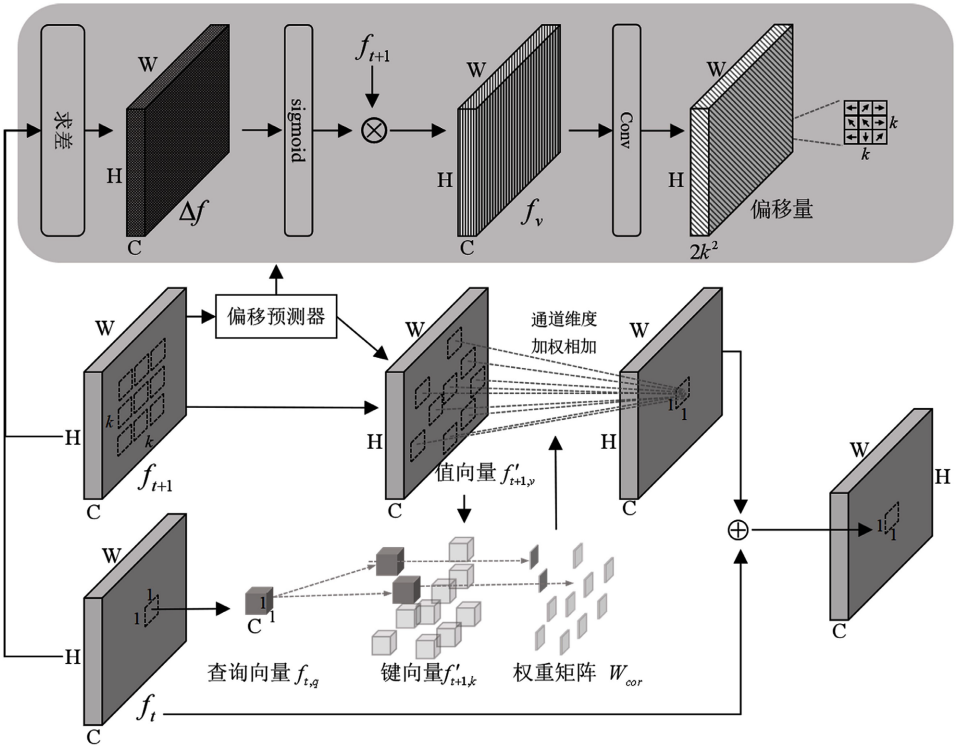

Figure 2. The interframe local deformable attention mechanism implements the details
图 2. 帧间局部可变形注意力机制实现细节

我们通过计算连续两个视频帧的特征图差异来获得时序差异线索，用两个视频帧的特征图差异来估计 $k \times k$ 网格内所有参考点的二维偏移量。特征图差异作为一种监督信号，指导每个参考点产生采样点。

具体来说，对于一对相邻的视频帧 x_t 和 x_{t+1} 的特征图 f_t 和 f_{t+1}，我们首先计算两个特征图的时序差异 Δf，公式如下：

$$\Delta f = f_{t+1} - f_t \tag{1}$$

然后，我们使用一个 sigmoid 函数对 Δf 进行操作，得到一个标准化的注意力图。这张注意力图包含着人物的注视关系变化趋势。因此，将注意力图 Δf 和第 $t+1$ 帧的特征图 f_{t+1} 相乘，获得注视关系变化趋势图(Gaze Variation Trend Map, GVTM) f_v，公式如下：

$$f_v = \text{sigmoid}(\Delta f) \times f_{t+1} \tag{2}$$

以注视关系变化趋势图(GVTM) f_v 为条件，我们利用偏移估计器来预测特征图 f_{t+1} 中以查询向量 Q_t 位置为中心的 $k \times k$ 局部区域中的每个空间位置 p 的二维偏移量 Δp。我们采用一个 2D 卷积实现偏移估计器，输出维度为 $2k^2$。

我们以特征图 f_t 的每个像素点作为查询向量 $f_{t,q} \in R^{C'}$，q 为查询向量的索引，并以 p 表示以查询向量 $f_{t,q}$ 为中心的 $k \times k$ 网格中的每个空间位置，p' 表示每个空间位置加上偏移量后的量，即 $p' = p + \Delta p$，Δp 是小数，所以 p' 是小数。我们通过双线性插值对每个不规则空间位置 p' 处的特征 $K'_{t+1}(p')$ 进行采样，公式如下：

$$f'_{t+1,k}(p')=\sum_{p}G(p,p')\cdot f_{t+1,k}(p) \tag{3}$$

其中，p 枚举了 $k\times k$ 网格中的所有整数空间位置，$f_{t+1,k}(p)$ 表示规则空间位置 p 处的特征，$G(\cdot)$ 为双线性插值算法。在对第 $t+1$ 帧的特征图 f_{t+1} 中所有的 k^2 个可变形特征进行采样后，我们将其作为第 t 帧的特征图 f_t 中每个查询向量 $f_{t,q}$ 的键向量 $f'_{t+1,k}\in\mathbb{R}^{C'\times\{k\times k\}}$ 和值向量 $f'_{t+1,v}\in\mathbb{R}^{C'\times\{k\times k\}}$。然后，我们通过公式 4，对相邻视频帧的局部可变形区域执行注意力机制，聚合这些可变形特征，提高每个查询向量 $f_{t,q}$ 的特征表示鲁棒性，进一步加强第 t 帧的特征图 f_t 的表示。

$$\begin{cases} W_{cor} = f_{t,q}\odot f'_{t+1,k} \\ A_{t+1}=W_{cor}\odot\left[f'_{t+1,v}\right]^{\mathrm{T}} \\ f'_{t,q} = f_{t,q}+A_{t+1} \end{cases} \tag{4}$$

4.2.2. 帧间可变形注意力机制

自注意力机制针对特征图上的所有像素点计算关联权重。为了解决这个问题，Deformable DETR [18]受可变形卷积[10]的启发，提出可变形注意力机制，即无论特征图的空间大小如何，都只关注给定的参考点周围的一小组关键采样点。可变形注意力机制通过计算参考点的偏移来得到采样点，偏移量的计算通过当前查询向量的线性映射得到。也就是说偏移量是通过输入特征的本身得到的，而没有利用到连续帧的时序线索。我们从 SIFA [17]中获得启发，提出帧间可变形注意力机制，如图所示，我们通过计算连续两个视频帧的特征图差异来获得时序线索，用两个视频帧的特征图差异来估计参考点的二维偏移量。特征图差异作为一种监督信号，指导参考点产生采样点，实现了动态采样，从而实现对于动态注视关系的时序建模。图 3 展示了我们提出的帧间可变形注意力机制的详细结构。

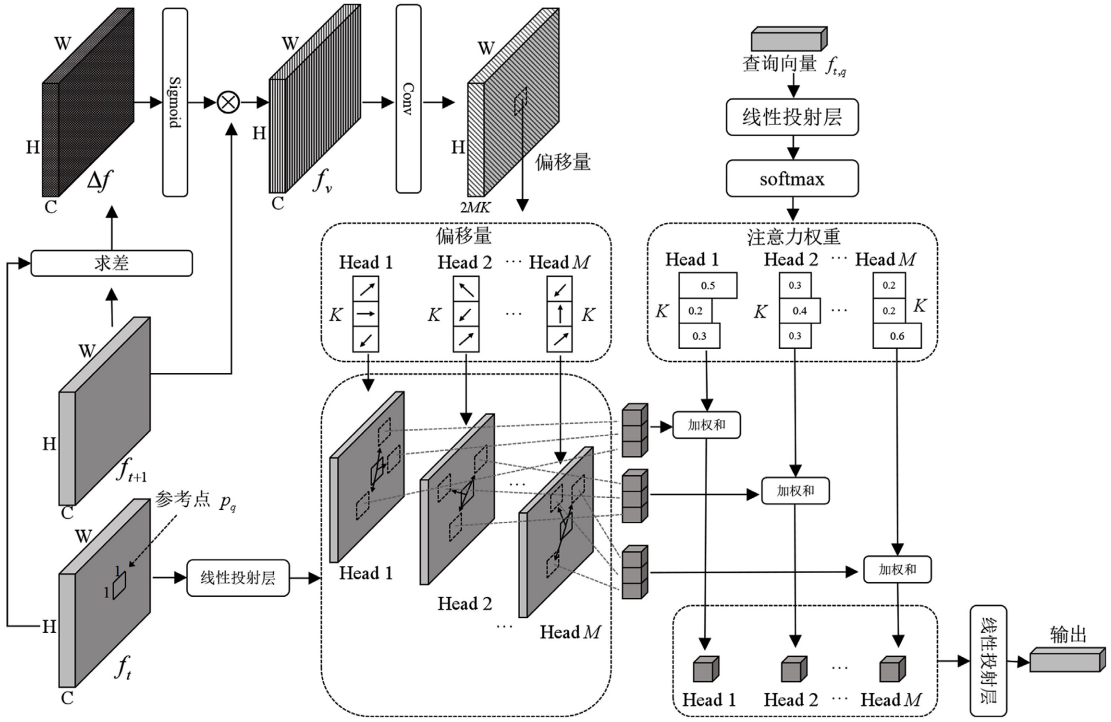

Figure 3. The interframe deformable attention mechanism implements the details
图 3. 帧间可变形注意力机制实现细节

具体来说，对于一对连续视频帧 x_t 和 x_{t+1} 的特征图 f_t 和 f_{t+1}，我们首先计算两个特征图的时序差异 Δf，公式如下：

$$\Delta f = f_{t+1} - f_t \tag{5}$$

然后，我们使用一个 sigmoid 函数对 Δf 进行操作，得到一个标准化的注意力图。这张注意力图包含着人物的注视关系变化趋势。因此，将注意力图 Δf 和第 $t+1$ 帧的特征图 f_{t+1} 相乘，获得注视关系变化趋势图(Gaze Variation Trend Map, GVTM) f_v，公式如下：

$$f_v = \text{sigmoid}(\Delta f) \times f_{t+1} \tag{6}$$

以注视关系变化趋势图(GVTM)作为输入，我们利用偏移估计器来预测所有参考点的位置 p 的二维偏移 Δp，并采用一个 2D 卷积实现偏移估计器，输出通道为 $2MK$。

具体来说，对于我们设计的帧间可变注意力机制，给定一对连续视频帧 x_t 和 x_{t+1} 的特征图 f_t 和 f_{t+1}，我们以特征图 f_t 的每个像素点作为查询向量 $f_{t,q} \in \mathbb{R}^{C'}$，$q$ 为查询向量的索引。p_q 表示参考点的二维坐标点，即查询向量 $f_{t,q}$ 对应的二维坐标点。根据标准 Transformer 对于多头自注意力[6]的设计，我们将帧间可变形注意力机制扩展到多头的形式，即帧间多头可变形注意力机制，用公式表示注意力特征的计算过程如下：

$$\text{In\_Frame\_DeformAtten}\left(f_{t,q}, p_q, f_t, f_{t+1}\right) = \sum_{m=1}^{M} W_m \left[\sum_{k=1}^{K} A_{mqk} \cdot W'_m f_t\left(p_q + \Delta p_{mqk}\right)\right] \tag{7}$$

上述公式中，m 表示多头注意力的头的索引，M 表示多头注意力的头数，k 表示采样点的索引，K 表示采样点的总数量。Δp_{mqk} 和 A_{mqk} 表示第 m 个注意头中第 k 个采样点的采样偏移量和注意力权重。$A_{mqk} \in [0,1]$ 是经过归一化后标量，表示在 $[0,1]$ 范围内的注意力权重，由 $\sum_{k=1}^{K} A_{mqk} = 1$ 进行标准化。我们将查询向量 $f_{t,q}$ 通过线性投影和归一化得到 A_{mqk}，输出通道为 MK。$\Delta p_{mqk} \in \mathbb{R}^2$ 是无约束的二维实数，所以 $p_q + \Delta p_{mqk}$ 是小数，我们使用双线性插值计算 $f_t\left(p_q + \Delta p_{mqk}\right)$。$W'_m \in \mathbb{R}^{C_v \times C'}$ 和 $W_m \in \mathbb{R}^{C' \times C_v}$ 都是可学习的权重矩阵，其中 $C_v = C'/M$。

4.2.3. 帧间可变形编码器(Inter-Frame Deformable Encoder, IFDE)

根据 Tu 等人提出的 HGTTR [14]，我们的帧间可变形编码器包含多个编码器层。如图所示，每个编码器层以帧间局部可变形注意力机制和帧间可变形注意力机制为核心，对于每个视频帧经过特征提取模块后得到的特征图 $f'_t \in \mathbb{R}^{C' \times H' \times W'}$，其相邻的下一个视频帧的特征图 f'_{t+1} 同时作为输入指导采样点的产生，我们首先对 f'_t 的空间维度进行展平操作，维度变为作为 $f'_t \in \mathbb{R}^{C' \times H'W'}$。由于 Transformer 结构与排列位置无关，我们添加一个固定的位置编码 $p \in \mathbb{R}^{C' \times H'W'}$ 用来补充位置信息。f'_t 和 p 相加后作为编码器的输入，并被学习为查询向量 Q_{en}、键向量 K_{en} 和值向量 V_{en}。

$$Q_{en} = W_{en,Q}\left(f'_t + p\right), K_{en} = W_{en,K}\left(f'_t + p\right), V_{en} = W_{en,V} f'_t \tag{8}$$

其中，$W_{en,Q} \in R^{C' \times C'}$，$W_{en,K} \in R^{C' \times C'}$，$W_{en,V} \in R^{C' \times C'}$ 都是可学习向量矩阵。我们将帧间可变形编码器的输出表示为 $f_{t,en} \in \mathbb{R}^{C' \times H'W'}$。

4.2.4. 空间解码器(Spatial Decoder, SD)

我们以标准的 Transformer 解码器结构构建我们的空间解码器，空间解码器包含多个解码器层。我们的空间解码器以自注意力机制和交叉注意力机制为核心。对于自注意力模块查询向量 Q_{des}、键向量 K_{des} 和值向量 V_{des} 用公式表示为：

$$\begin{cases} Q_{des} = W_{des,Q}\left(Q_{des,f} + Q_{des,p}\right) \\ K_{des} = W_{des,K}\left(K_{des,f} + Q_{des,p}\right) \\ V_{des} = W_{des,V} V_{des,f} \\ Q_{des,f} = K_{des,f} = V_{des,f} \\ Q_{des,p} = \text{Embedding}\left(N_q, C'\right) \end{cases} \tag{9}$$

其中，$Q_{des,f} \in \mathbb{R}^{C' \times N_q}$ 是我们在解码器的第一层初始化的一个常数向量，$Q_{des,p} \in \mathbb{R}^{C' \times N_q}$ 是可学习的位置编码，我们将其称之为注视关系查询向量，其中每一个注视关系查询元素对应一个注视关系实例(定义见第3节)。和编码器类似，我们将其添加到每个注意层的输入中。N_q 是一个常数，其大小设置远大于任意一个视频中所有的注视关系实例的数量，$W_{des,Q}$，$W_{des,K}$，$W_{des,V}$ 都是可学习向量矩阵。

对于交叉注意力模块，模块查询向量 Q_{dec}、键向量 K_{dec} 和值向量 V_{dec} 用公式表示为：

$$\begin{cases} Q_{dec} = W_{dec,Q}\left(Q'_{dec,f} + Q_{dec,p}\right) \\ K_{dec} = W_{dec,K}\left(f_{t,en} + p\right) \\ V_{dec} = W_{dec,V} f_{t,en} \end{cases} \tag{10}$$

其中，$Q'_{dec,f} \in \mathbb{R}^{C' \times N_q}$ 是自注意模块的输出，$Q_{dec,p}$ 的设置和公式(9)中的 $Q_{des,p}$ 一样。$W_{dec,Q}$，$W_{dec,K}$，$W_{dec,V}$ 都是可学习向量矩阵。

总的来说，空间解码器有三个输入：来自帧间可变形编码器的输出特征，注视关系查询集合和可学习的位置编码。空间编码器将 N_q 个注视关系查询转换为一个输出嵌入，我们将空间解码器的输出表示为 $f_{t,de} \in \mathbb{R}^{C' \times N_q}$。

4.3. 时序 Transformer

时序 Transformer 包含三个组成部分：时序注视关系特征编码器用来融合来自所有的帧间可变形编码器输出的特征，形成时序注视关系特征，作为时序注视关系查询的空间线索。时序注视关系查询编码器沿时间维度连接所有的空间解码器输出的注视关系查询向量。时序注视关系解码器以时序注视关系查询编码器和时序注视关系特征编码器的输出为输入，解码产生当前视频帧的最终的时序注视关系查询向量，用于预测模块产生最终的预测结果。

4.3.1. 时序注视关系查询编码器(Temporal Gaze Query Encoder, TGQE)

如前一部分所述，可学习的注视关系查询在训练过程中自动学习整个视频帧的空间上下文特征。这意味着注视关系查询与不相邻视频帧的时间上下文无关。因此，我们提出了一个简单而有效的编码器来建模当前帧中的注视关系查询和参考帧中的注视关系查询之间的相互作用。

我们的核心思想是通过一个编码器将每个视频帧通过帧间可变形 Transformer 建模得到的注视关系查询连接起来，从而学习整个视频帧序列的时间上下文信息。我们将此模块命名为时序注视关系查询编码器。该编码器通过编码当前视频帧与所有参考视频帧中所有的注视关系查询，以增强当前帧的注视关系查询，它输出融合时序信息后的时序注视关系查询。

根据第3节中的设置，对于视频帧序列 X 中的每个视频帧 $x_t \in \mathbb{R}^{3 \times H \times W}$，$t \in \{t_0 - i, \cdots, t_0 - 1, t_0, t_0 + 1, \cdots, t_0 + i\}$，我们将其经过帧间可变形 Transformer 后得到的注视关系查询向量 $f_{t,de} \in \mathbb{R}^{C' \times N_q}$ 表示为 $q_t \in \mathbb{R}^{1 \times \left(C' \cdot N_q\right)}$，并将其构成序列 $I \in \mathbb{R}^{(2i+1) \times \left(C' \cdot N_q\right)}$。将序列 I 中的每个 q_t 通过可学习的线性映射层，得到一个具有可学习性的特征表达。时序注视关系查询编码器利用时序位置编码来保留时间序列的位置信息。可表述为：

$$Z=\mathrm{Concat}\left(q_{t_0-i}E,\cdots,q_{t_0-1}E,q_{t_0}E,q_{t_0+1}E,\cdots,q_{t_0+i}E\right)+E_{pos} \tag{11}$$

其中，$E\in\mathbb{R}^{\left(C'\cdot N_q\right)\times C''}$ 表示线性映射层中可学习映射矩阵，通过线性映射层后与时序位置编码 $E_{pos}\in\mathbb{R}^{(2i+1)\times C''}$ 相加，输入序列 I 变为 $Z\in\mathbb{R}^{(2i+1)\times C''}$，其中 C'' 为线性映射后的维度。Z 构成了时序注视关系查询编码器的输入。通过不同的可学习线性变化作为自注意力模块的查询向量、键向量和值向量。

时序注视关系查询编码器的输出 $Y\in\mathbb{R}^{f\times C''}$ 和输入 $Z\in\mathbb{R}^{f\times C''}$ 有着相同的特征维度大小。因为我们通过编码当前视频帧与所有参考视频帧中所有的注视关系查询，以增强当前帧的注视关系查询，得到的编码器的输出 Y 需要通过取帧维度中的平均值被压缩到向量 $y\in\mathbb{R}^{1\times C''}$，然后通过一个使用层归一化的多层感知机模块，将结果回归到 $y\in\mathbb{R}^{1\times\left(C'\cdot N_q\right)}$，最后调整 y 的维度，将其表示为 $f_{t_0,tqen}\in\mathbb{R}^{C'\times N_q}$，用来表示融合后的中间帧的注视关系查询，我们将其成为时序注视关系查询向量。

4.3.2. 时序注视关系特征编码器(Temporal Gaze Feature Encoder, TGFE)

受到 Zhou 等人提出 TransVOD [27]的启发，我们使用时序可变形注意力机制构建一个编码器，用于融合当前帧和参考帧的注视关系特征 $f_{t,en}\in\mathbb{R}^{C'\times H'W'},t\in T$，并输出增强的时序注视关系特征 $f_{t_0,tfen}$，我们将其称之为时序注视关系特征编码器。编码器包含自注意力机制和时序可变形注意力机制，自注意力机制为标准 Transformer 自注意力模块。对于多头时序可变形注意力机制，表示如下：

$$\mathrm{TemDeformAtten}\left(f_{t_0,en}^{q},\hat{p}_q,\left\{f_{t.en}\right\}_{t\in T}\right)=\sum_{m=1}^{M}W_m\left[\sum_{t\in T}\sum_{k=1}^{K}A_{mtqk}\,f_{t,en}^{q}\left(\phi_t\left(\hat{p}_q\right)+\Delta p_{mtqk}\right)\right] \tag{12}$$

上述公式中，q 和 k 分别表示查询和键元素的索引，我们以特征图 $f_{t_0,en}$ 中的每个像素点作为查询向量 $f_{t_0,en}^{q}$。p_q 表示参考点的二维坐标，即查询向量 $f_{t_0,en}^{q}$ 对应的二维坐标点。$\hat{p}_q\in[0,1]^2$ 为 p_q 归一化坐标，坐标 $(0,0)$ 代表左上角，$(1,1)$ 代表右下角。m 表示多头注意力头的索引，M 表示多头注意力的头数，t 表示视频帧的索引，T 表示视频帧索引的集合，k 表示采样点的索引，K 表示在所有帧特征图上的采样点总数量。Δp_{mtqk} 和 A_{mtqk} 表示第 m 个注意力头中第 t 帧的第 k 个采样点的采样偏移量和注意力权重。$A_{mtqk}\in[0,1]$ 是经过归一化后标量，由 $\sum_{t\in T}\sum_{k=1}^{K}A_{mtqk}=1$ 标准化。$\Delta p_{mtqk}\in\mathbb{R}^2$ 是无约束范围的二维实数，我们将查询特征 $f_{t_0,en}^{q}$ 通过线性投影得到 Δp_{mtqk} 和 A_{mtqk}。因为 $p_q+\Delta p_{mlqk}$ 是小数，我们使用双线性插值计算 $f_{t,en}^{q}\left(\phi_t\left(\hat{p}_q\right)+\Delta p_{mtqk}\right)$。$\phi_t\left(\hat{p}_q\right)$ 表示将归一化坐标 $\hat{p}_q$ 重新扩展到第 t 帧的特征图。

4.3.3. 时序注视关系解码器(Temporal Gaze Decoder, TGD)

我们以标准的 Transformer 解码器结构构建我们的时序注视关系解码器，其包含多个解码器层。我们的时序注视关系解码器以自注意力机制和交叉注意力机制为核心。注意力模块以时序注视关系特征编码器的输出 $f_{t_0,tfen}$ 为输入，用于学习查询向量、键向量和值向量。

交叉注意力的输入有两部分：由自注意力模块的输出学习键向量和值向量，由时序注视关系查询编码器的输出 $f_{t_0,tqen}$ 学习为查询向量。时序解码器的输出表示为 $f_{t_0,trde}\in\mathbb{R}^{C'\times N_q}$，用于预测模块的输入。

4.4. 预测模块

预测模块将时序解码器的解码输出的时序注视关系查询向量分解到每个注视关系查询通道中，以描述每一对注视关系。给定注视关系查询通道 r，使用特征 $f_{t_0,trde}^{r}\in\mathbb{R}^{C'\times 1}$ 学习头部标签 p_h^r、头部位置 l_h^r、观看外部标签 p_g^r 和注视目标位置热图 l_g^r。

具体来说，我们分别使用两个由一层全连接层和 softmax 函数构成的多层感知机来预测头部置信度得分 p_h^r 和注视位置(帧内/帧外) p_g^r。同时，设置一个由三层全连接层和 sigmoid 函数构成的多层感知机来预测头部边界框 l_h^r，以及一个由五层全连接层和 sigmoid 函数构成的多层感知机来预测注视目标位置热

图 p_g^r，表示如下：

$$\begin{cases} p_h^r = \text{softmax}\left(\text{MLP}\left(f_{t_0,trde}^r\right)\right) \\ p_g^r = \text{softmax}\left(\text{MLP}\left(f_{t_0,trde}^r\right)\right) \\ l_h^r = \text{sigmoid}\left(\text{MLP}\left(f_{t_0,trde}^r\right)\right) \\ l_g^r = \text{reshape}\left(\text{sigmoid}\left(\text{MLP}\left(f_{t_0,trde}^r\right)\right)\right) \end{cases} \tag{13}$$

4.5. 损失函数

原始的 DETR 为了避免后处理的过程，对每个对象查询的解码器的输出特征通过 FFN 进一步转换，以输出每个对象的类别得分和边界框位置。给定边界框和类别得分，在预测值和真实值之间应用匈牙利算法，为每个对象查询进行一对一匹配。我们遵循 DETR 的训练方式，并使用匈牙利算法算法进行二部匹配，匹配完成后，再按照匹配对计算损失和。

4.5.1. 二分匹配

在我们的模型输入设置中，N_q 是足够大的，具体可见第 5 节的消融实验，数据集中所有视频帧中的注视关系实例均小于 N_q，我们用 $\varnothing$ (无注视关系实例)填充真实标签，使所有的真实标签的注视关系实例集合的大小变为 N_q。

如图中所示，模型输出一组大小固定的数据 N_q 表示预测的注视关系实例。我们将这些数据表示为 $O=o^i, i=1,2,\cdots,N_q$。同时，我们将真实值表示为：$T=t^i, i=1,2,\cdots,M,\cdots,\varnothing_1,\varnothing_2,\cdots,\varnothing_{N_q-M}$，其中 M 表示一个视频帧中真实的注视关系实例的数量。所以，匹配的过程可以表示为：$\omega_{T\to O}$，i 表示真实值的索引，$\omega(i)$ 表示匹配到的第 i 个真实值的预测值的索引。我们定义匹配权重函数为：

$$\begin{cases} L_{cost} = \sum_i^{N_q} L_{match}\left(t^i, o^{\omega(i)}\right) \\ L_{match}\left(t^i, o^{\omega(i)}\right) = \beta_1 L_{box} + \beta_2 L_h + \beta_3 L_w + \beta_4 L_g \end{cases} \tag{14}$$

$L_{match}\left(t^i, o^{\omega(i)}\right)$ 是第 i 个真实值和第 $\omega(i)$ 个预测值之间的匹配损失。匹配损失 $L_{match}\left(t^i, o^{\omega(i)}\right)$ 包含四个部分：头部分类损失 L_h 使用二分类交叉熵损失、头部边界框损失 L_{box} 由 L_1 损失和 GIoU 损失构成、帧外注视损失 L_w 使用二分类交叉熵损失、注视目标热图损失 L_g 使用 L_2 损失，公式表示如下：

$$\begin{cases} L_h = -\left(t_h^i \cdot \log\left(o_h^{w(i)}\right) + \left(1-t_h^i\right)\log\left(1-o_h^{w(i)}\right)\right) \\ L_{box} = \alpha_1 \left\| t_b^i - o_b^{w(i)} \right\| - \alpha_2 \text{GIoU}\left(t_b^i - o_b^{w(i)}\right) \\ L_w = -\left(t_w^i \cdot \log\left(o_w^{w(i)}\right) + \left(1-t_w^i\right)\log\left(1-o_w^{w(i)}\right)\right) \\ L_g = \left\| t_g^i - o_g^{w(i)} \right\|_2 \end{cases} \tag{15}$$

其中，i 表示真实值的索引，$\omega(i)$ 表示匹配到的第 i 个真实值的预测值的索引。t_h^i 表示头部标签的真实值，t_b^i 表示头部边界框的真实值，t_g^i 表示注视目标的真实值，t_w^i 表示帧外注视概率的真实值。$o_h^{w(i)}$ 表示头部标签的预测值，$o_b^{w(i)}$ 表示头部边界框的预测值，$o_w^{w(i)}$ 表示帧外注视概率的预测值，$o_g^{w(i)}$ 表示预测的注视目标热图中最大值点的坐标。

然后，我们利用匈牙利算法来确定所有的一一匹配关系，以达到最优分配。

4.5.2. 损失函数

在找到真实值和预测值之间的最优的一对一匹配后，计算损失函数公式同公式(15)所示，超参数的设置同公式(14)。

5. 实验

5.1. 数据集和评价指标

5.1.1. 数据集介绍

我们在 VideoAttentionTarget [5]和 VideoCoAtt [8]两个数据集上定量的评估了我们提出的模型的性能。

VideoAttentionTarget 数据集由 Youtube 上各种视频来源的 1331 个视频剪辑组成。VideoAttentionTarget 的注释包括 164,541 个帧级头部边界框，109,574 个帧内凝视目标，54,967 个帧外凝视目标。测试集包含 31,978 个标签，其余用于训练集。我们使用 VideoAttentionTarget 数据集评估我们的模型在视频注视目标检测任务中的性能。

VideoCoAtt 数据集是从 20 个不同的电视节目或电影中精心收集的 380 个 RGB 视频序列构成的。每个视频序列持续的时间不同，从 20 秒左右到 1 分钟以上，帧率为 25 fps。总共有 492,100 个视频帧，数据集标注了人物头部边界框及注视目标边界框，由于数据集是用于研究共同注视目标的，我们使用 VideoCoAtt 数据集评估我们的模型在视频中共同注视目标检测任务中的性能。

5.1.2. 评价指标

因为 VideoAttentionTarget [5]和 VideoCoAtt [8]两个数据集的标注略有不同，所以在两个数据集上使用的的评价指标略有不同，如下所述：

对于 VideoAttentionTarget 数据集，数据集标注给出了每个人物的头部边界框(由左上角坐标和右下角坐标确定出唯一的框)和注视点的位置(单点坐标)。我们采用标准的评价指标来评估我们模型的性能：AUC 和 L_2 距离(欧氏距离)。AUC↑：我们使用曲线下面积(AUC)标准来评估预测热图的置信度。↑表示 AUC 值越大，模型预测精度越高。L_2 距离↓：我们使用注视目标位置热图中最大值所对应的像素点坐标作为预测的注视目标坐标，以此来计算预测值与真实注视点之间的 L_2 距离，我们将所有坐标点按照视频帧分辨率大小进行归一化，↓表示 L_2 距离越小，模型的预测越准确。此外，我们还使用 AP↑来评价模型在检测注视目标位置在该视频帧外还是帧内任务上性能，↑表示 AP 值越大，模型预测精度越高。

对于 VideoCoAtt 数据集，数据集标注给出了每个人物的头部边界框和共同注视目标的边界框。我们使用 L_2 距离(欧氏距离)来评价模型在检测共同注视目标任务上的性能。我们使用注视目标位置热图中最大值所对应的像素点坐标作为预测的注视目标坐标，使用注视目标边界框的中心点作为真实注视目标坐标，用两者来计算 L_2 距离。除此之外，我们使用共同注视预测精度(Acc)来评价模型在检测共同注视帧上的性能。Acc↑：预测结果中存在共同注视的视频帧占所有标注的存在共同注视的视频帧的比例作为预测精度。↑表示 Acc 值越大，模型预测精度越高。存在共同注视目标的定义为：一个视频帧中存在两个及两个以上人物在注视同一个目标时，此视频帧为存在共同注视的视频帧。

5.2. 实现细节

我们用 PyTorch 实现了我们的模型。每个视频帧在输入时，尺寸被调整为 224 × 224。根据 Tu 等人[14]对于 HGTTR 的设计，我们使用 ResNet-50 [26]作为我们的骨干网络用来提取视频帧的特征。对于帧间可变形 Transformer，我们遵循 DETR [23]的设计，使用具有 6 层 8 头 256 维的编码器和具有 6 层 8 头 256 维的解码器。我们设置时序注视关系查询编码器的层数为 3，时序注视关系特征编码器的层数为 1，两者的输入帧数均使用 5 帧(即参考帧数量为 4)。时序解码器的层数为 1。输出注视目标位置热图的分辨

率为[64, 64]。我们通过设置高斯中心作为地面真值凝视目标的位置，并设置高斯标准差为 3 像素来提供地面真值凝视目标热图。我们将主干网络的学习率设为 2×10^{-5}，Transformer 的学习率为 2×10^{-4}，权重衰减设为 1×10^{-4}，使用 AdamW 作为优化器，批次大小为 16。所有 Transformer 结构的参数都是用 Xavier 初始化。注视关系查询数量 N_q 为 20，匹配损失函数中的权值 $\beta_1, \beta_2, \beta_3, \beta_4$ 分别设置为 2，1，1，2。我们使用在 ImageNet 数据集上预训练的 ResNet-50 参数初始化我们的主干网络。在训练方式上，为了模型更快的收敛，我们首先对帧间可变形 Transformer 训练 80 个 epoch，然后保持此部分参数不变，对剩下的模块训练 40 个 epoch。

5.3. 模型的整体性能

5.3.1. 视频注视目标检测

如表 1 所示，我们在 VideoAttentionTarget 数据集上用我们的模型和之前的方法进行了定量的对比实验。需要注意的是，由于基于卷积的方法需要裁剪后的头部图像和头部位置，所以我们将对比数据分为了“HeadGT”和“ExHead”两种情况下。其中，“HeadGT”表示直接使用数据集标签中给出的真实的头部框的位置作为输入，“ExHead”表示使用一个额外的头部位置检测器。根据[5]，我们在 VideoAttentionTarget 数据集上微调了一个基于 SSD 的头部检测网络。

Table 1. Experimental results of the model on the VideoAttentionTarget dataset
表 1. 模型在 VideoAttentionTarget 数据集上的实验结果

对比方法	注视目标位置在帧内的情况				在帧外的情况	
	AUC↑		L_2 距离↓		AP↑	
	HeadGT	ExHead	HeadGT	ExHead	HeadGT	ExHead
Random [2]	0.505	0.247	0.458	0.592	0.621	0.349
Fixed bias [2]	0.728	0.522	0.326	0.472	0.624	0.510
Chong [3]	0.830	0.791	0.193	0.214	0.705	0.651
Bao [4]	0.885	-	0.120	-	0.869	-
Chong [5]	0.860	0.812	0.134	0.146	0.853	0.849
Lian [6]	0.837	0.784	0.165	0.172	-	-
Miao [11]	0.917	-	0.109	-	0.908	-
Fang [12]	0.905	-	0.108	-	0.896	-
Jin [13]	0.90	-	0.104	-	0.895	-
Tonini [16]	0.940	0.894	0.129	0.182	-	-
HGTTR [14]	-	0.893	-	0.137	-	0.821
GOT [15]	-	0.923	-	0.102	-	**0.944**
GOT + Depth [15]	-	0.933	-	0.104	-	0.934
Ours	-	**0.935**		**0.093**		0.923

根据表 1 中的实验数据，可以看出我们的模型在此数据集上优于之前的所有方法。我们将之前的方法分为两大类：基于卷积提取注视特征的方法和基于 Transformer 提取注视特征的方法。对于基于卷积提取注视特征的方法，Chong 等人[5]提出的 VideoAttention 除了考虑头部姿态、场景显著信息，额外使用

Conv-LSTM 融合时序信息。我们的方法提取了相邻帧间和整个序列两个维度的时序信息，提高了模型对于动态注视关系的建模能力，也取得了更好的性能表现。另外，Tu 等人提出 HGTTR [14]首次使用 Transformer 结构提取注视关系特征，GOT 在此基础上加入了目标感知注意力，提升了模型在静态场景下的表现，但是没有结合视频数据集中的时序信息，因此我们的模型在帧内注视目标的检测指标上表现均优于两者。

5.3.2. 视频共同注视目标检测

我们的方法旨在同时检测视频帧中所有人物的注视目标，因此模型在本质上非常适合用于推断社交场景中的共同注视目标。如表 2 所示，我们在 VideoCoAtt 数据集上用我们的模型和之前的方法进行了定量的对比实验。从实验结果可以看出，我们的方法在两项评价指标上均优于之前提出的所有方法。这证明，我们的模型具有在社交场景中识别高水平的人类注视关系的能力。

Table 2. Experimental results on the VideoCoAtt dataset
表 2. 在 VideoCoAtt 数据集上的实验结果

对比方法	L_2 距离↓	Acc↑
Random [2]	286	50.8
Fixed bias [2]	122	52.4
GazeFollow [2]	102	58.7
Chong [5]	57	83.3
Fan [8]	62	71.4
HGTTR [14]	46	90.4
Deep Convnet [26]	83	59.4
Deep Convnet [26] + LSTM	71	66.2
Sumer [27]	63	78.1
Ours	**41**	**92.5**

5.4. 消融实验

为了证明我们提出的方法中关键模块的效果，我们进行了大量的实验，用来研究这些模块对于整体模型的影响。如无特别说明，均以 ResNet-50 作为骨干网络在 VideoAttentionTarget 数据集上进行消融实验。

5.4.1. 模型的各个组成模块的影响

表 3 展示了我们模型各个组成模块的有效性。帧间可变形 Transformer、时序注视关系查询编码器、时序注视关系特征编码器和时序注视关系解码器是我们提出的四个关键模块。方法 A 为 Tu 等人提出的 HGTTR，与之相对比的是方法 B，仅使用帧间可变形 Transformer。由于我们使用相邻两帧的特征差异，指导产生动态采样点，从而实现对于动态注视关系的建模，取得了一定的性能提升。方法 C 在 B 的基础上添加时序注视关系查询编码器，通过对视频帧序列的所有视频帧输出的注视关系查询向量进行融合，增强了当前帧的注视关系查询向量的表达性，取得了了较大的性能提升。方法 D 使用时序注视关系特征编码器和时序注视关系解码器，可以看出性能弱于方法 C，说明时序注视关系查询编码器的作用更大。方法 E 使用时序注视关系查询编码器和时序注视关系解码器，相对方法 B 取得了更好的结果。实验结果表明，我们所提出的这些模块，对最终的模型性能提升都是有必要的。

Table 3. The ablation results were obtained for each component module of the model
表 3. 对模型的各个组成模块进行消融结果

方法	HGTTR [18]	IFTR	TGQE	TGFE	TGD	AUC↑	L_2 距离↓	AP↑
A	√					0.869	0.137	0.812
B		√				0.883	0.128	0.853
C		√	√			0.915	0.109	0.913
D		√		√	√	0.892	0.117	0.903
E		√	√		√	0.921	0.103	0.917
F		√	√	√	√	0.935	0.093	0.923

5.4.2. 参考视频帧的数量对模型性能的影响

表 4 展示了对于参考视频帧数量的消融研究。当参考帧数量大于 1 时，实验前提设置时序注视关系查询编码器为 1 个编码器层，时序注视关系特征编码器为 1 个编码器层，时序注视关系解码器为 1 个解码器层。实验结果表明，当参考帧数增加时，三项评价指标均会随着提高。但是当参考帧数达到 4 时，模型的性能将趋于稳定，综合考虑，我们在最终的模型中将参考帧的数量，设置为 4，即视频帧序列的长度为 5。

Table 4. The influence of the number of different reference frames on the experimental results
表 4. 不同参考帧的数量对实验结果的影响

参考帧数量 $2i$	当前帧	帧序列数量 $2i+1$	AUC↑	L_2 距离↓	AP↑
0	1	1	0.869	0.137	0.812
2	1	3	0.894	0.118	0.883
4	1	5	0.927	0.106	0.914
6	1	7	0.926	0.102	0.913

5.4.3. 帧间注视关系查询编码器的层数影响

我们对帧间注视关系查询编码器层数的消融实验结果在表 5 中呈现。实验前提设置参考帧数量为 4，时序注视关系特征编码器层数为 1，时序注视关系解码器层数为 1。实验结果说明，当层数设置为 3 时，效果最好。当层数再增加时，性能基本保持不变。

Table 5. The effect of the number of layers of TGQE
表 5. 帧间注视关系查询编码器的层数影响

层数	1	2	3	4	5	6
AUC↑	0.927	0.932	0.935	0.936	0.935	0.936
L_2 距离↓	0.106	0.097	0.093	0.096	0.095	0.096
AP↑	0.914	0.920	0.923	0.924	0.923	0.926

5.4.4. 帧间注视关系特征编码器层数的影响

我们对帧间注视关系特征编码器中编码器层数的消融实验结果在表 6 中呈现。实验前提设置为参考帧数量为 4，时序注视怪查询编码器的层数为 1，时序注视关系解码器层数为 1。实验结果表明，当编码

器层的数量设置为 1 时效果最好，这意味着更多的编码器层数不会对最终性能带来任何好处。实验说明了通过时序可变形注意力模块将特征聚集在一个时间维度上，对于学习视频帧序列中的时间上下文信息是有用的。

Table 6. The effect of the number of layers of TGFE
表 6. 帧间注视关系特征编码器的层数影响

层数	1	2	3	4	5	6
AUC↑	**0.927**	0.927	0.926	0.927	0.928	0.928
L_2 距离↓	**0.106**	0.108	0.108	0.106	0.105	0.106
AP↑	**0.914**	0.912	0.916	0.909	0.913	0.915

5.4.5. 时序注视关系解码器层数的影响

表 7 说明了对时序注视关系解码器中解码器层数的消融研究。实验设置的基本前提为：参考视频帧的数量为 4 个，帧间注视关系查询编码器层数为 1，帧间注视关系特征编码器层数为 1。实验结果表明，在 TDTD 中只需要一个解码器层。

Table 7. The effect of the number of layers of TGD
表 7. 时序注视关系解码器的层数影响

层数	1	2	3	4	5	6
AUC↑	0.927	0.921	0.924	0.924	0.926	0.927
L_2 距离↓	0.106	0.104	0.108	0.110	0.107	0.109
AP↑	0.914	0.914	0.916	0.913	0.914	0.916

5.4.6. 注视关系查询集合大小 N_q 的影响

我们探究了不同大小的注视关系查询集合对模型性能的影响。如表 8 所示，当集合中注视关系查询的数量设置为 30 时效果最好，但在超过 20 时，模型的性能基本保持不变。综合考虑，我们将实例的数量设置为 20。

Table 8. The effect of gaze query set size
表 8. 注视关系查询集合大小的影响

实例数量	10	15	20	25	30	35
AUC↑	0.896	0.918	0.927	0.927	0.928	0.927
L_2 距离↓	0.129	0.116	0.106	0.105	0.105	0.106
AP↑	0.893	0.901	0.914	0.913	0.914	0.915

5.5. 可视化结果

如图 4、图 5 所示，两组实验均是在 VideoAttentionTarget 数据集上的结果。其中，图 5 主要表现了我们的模型在运动模糊情况下的表现。可以看出在第二帧和第三帧中出现了较严重的运动模糊，第二帧中单帧模型对于注视目标检测出现较大误差，而我们的模型检测更为精准。单帧模型对于第三帧中人物头部的检测出现较大误差，而我们的模型对于头部边界框的预测更接近真实值。图 5 主要表现了我们的模型在动态注视情况下的表现。可以看出从第一帧到第二帧，图中人物的注视目标有着较大的变化，但

是其头部姿态的变化较小，单帧模型对于第 2、3、4 帧的检测均出现较大的误差，而我们的模型由于结合了时序变化信息，并且具有较长的时序信息捕获能力，预测的结果也更加精确。

Figure 4. Performance of the model in the case of motion ambiguity
图 4. 模型在运动模糊情况下的表现

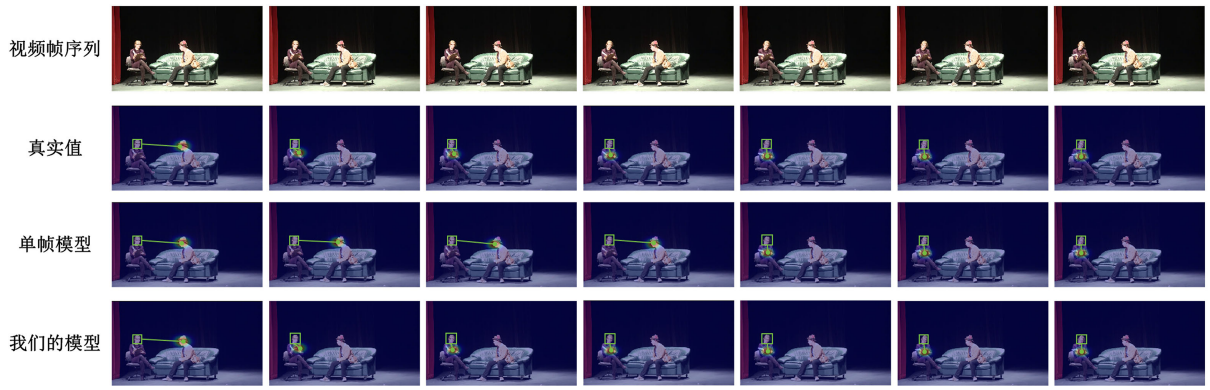

Figure 5. The performance of the model under dynamic gaze
图 5. 模型在动态注视情况下的表现

6. 结束语

在这篇论文中，我们提出了一种新颖的端到端的视频注视目标检测模型，通过利用帧间局部可变形注意力、帧间可变形注意力和时序 Transformer 实现了空间、相邻帧间和视频帧序列三个维度的建模，从而提高了对空间特征和时序特征的表达能力。我们的核心思想是通过帧间局部可变形注意力提高对于空间特征的表示，通过帧间可变形注意力提高对动态注视关系的表示能力，通过时序 Transformer 在每一帧中聚合空间注视关系查询和注视关系特征。这些设计通过显著的提升了在视频数据上对人物注视目标的检测性能。我们还进行了大量实验来验证核心组件的有效性。我们在 VideoAttentionTarget 和 VideoCoAtt 两个公开数据集中取得了最好的性能，并且我们的工作是第一个将 Transformer 应用于视频注视目标检测任务。

参考文献

[1] Judd, T., Ehinger, K., Durand, F., *et al.* (2009) Learning to Predict Where Humans Look. 2009 *IEEE* 12*th International Conference on Computer Vision*, Kyoto, 29 September-02 October 2009, 2106-2113. https://doi.org/10.1109/ICCV.2009.5459462

[2] Recasens, A., Khosla, A., Vondrick, C., *et al.* (2015) Where Are They Looking? *Advances in Neural Information Processing Systems*, **28**, 199-207.

[3] Chong, E., Ruiz, N., Wang, Y., *et al.* (2018) Connecting Gaze, Scene, and Attention: Generalized Attention Estimation via Joint Modeling of Gaze and Scene Saliency. In: Ferrari, V., Hebert, M., Sminchisescu, C. and Weiss, Y., Eds., *Computer Vision—ECCV* 2018, *Lecture Notes in Computer Science*, Vol. 11209, Springer, Cham, 383-398. https://doi.org/10.1007/978-3-030-01228-1_24

[4] Bao, J., Liu, B. and Yu, J. (2022) Escnet: Gaze Target Detection with the Understanding of 3d Scenes. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, New Orleans, 18-24 June 2022, 14126-14135. https://doi.org/10.1109/CVPR52688.2022.01373

[5] Chong, E., Wang, Y., Ruiz, N., *et al.* (2020) Detecting Attended Visual Targets in Video. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Seattle, 13-19 June 2020, 5396-5406. https://doi.org/10.1109/CVPR42600.2020.00544

[6] Lian, D., Yu, Z. and Gao, S. (2018) Believe It or Not, We Know What You Are Looking at! In: Jawahar, C., Li, H., Mori, G. and Schindler, K., Eds., *Computer Vision—ACCV* 2018, *Lecture Notes in Computer Science*, Vol. 11363, Springer, Cham, 35-50. https://doi.org/10.1007/978-3-030-20893-6_3

[7] Recasens, A., Vondrick, C., Khosla, A., *et al.* (2017) Following Gaze in Video. *Proceedings of the IEEE International Conference on Computer Vision*, Venice, 22-29 October 2017, 1435-1443. https://doi.org/10.1109/ICCV.2017.160

[8] Fan, L., Chen, Y., Wei, P., *et al.* (2018) Inferring Shared Attention in Social Scene Videos. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Salt Lake City, 18-23 June 2018, 6460-6468. https://doi.org/10.1109/CVPR.2018.00676

[9] Zhou, Q., Li, X., He, L., *et al.* (2022) TransVOD: End-to-End Video Object Detection with Spatial-Temporal Transformers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45, 7853-7869. https://doi.org/10.1109/TPAMI.2022.3223955

[10] Dai, J., Qi, H., Xiong, Y., *et al.* (2017) Deformable Convolutional Networks. *Proceedings of the IEEE International Conference on Computer Vision*, Venice, 22-29 October 2017, 764-773. https://doi.org/10.1109/iccv.2017.89

[11] Miao, Q., Hoai, M. and Samaras, D. (2023) Patch-Level Gaze Distribution Prediction for Gaze Following. *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, Waikoloa, 2-7 January 2023, 880-889. https://doi.org/10.1109/WACV56688.2023.00094

[12] Fang, Y., Tang, J., Shen, W., *et al.* (2021) Dual Attention Guided Gaze Target Detection in the Wild. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Nashville, 20-25 June 2021, 11390-11399. https://doi.org/10.1109/CVPR46437.2021.01123

[13] Jin, T., Yu, Q., Zhu, S., *et al.* (2022) Depth-Aware Gaze-Following via Auxiliary Networks for Robotics. *Engineering Applications of Artificial Intelligence*, **113**, Article 104924. https://doi.org/10.1016/j.engappai.2022.104924

[14] Tu, D., Min, X., Duan, H., *et al.* (2022) End-to-End Human-Gaze-Target Detection with Transformers. 2022 *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*), New Orleans, 18-24 June 2022, 2192-2200. https://doi.org/10.1109/CVPR52688.2022.00224

[15] Tonini, F., Dall'Asen, N., Beyan, C., *et al.* (2023) Object-Aware Gaze Target Detection. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, Paris, 1-6 October 2023, 21860-21869. https://doi.org/10.1109/ICCV51070.2023.01998

[16] Tonini, F., Beyan, C. and Ricci, E. (2022) Multimodal across Domains Gaze Target Detection. *Proceedings of the* 2022 *International Conference on Multimodal Interaction*, Bengaluru, 7-11 November 2022, 420-431. https://doi.org/10.1145/3536221.3556624

[17] Long, F., Qiu, Z., Pan, Y., *et al.* (2022) Stand-Alone Inter-Frame Attention in Video Models. 2022 *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*), New Orleans, 18-24 June 2022, 3192-3201. https://doi.org/10.1109/CVPR52688.2022.00319

[18] Zhu, X., Su, W., Lu, L., *et al.* (2020) Deformable DETR: Deformable Transformers for End-to-End Object Detection.

[19] Saran, A., Majumdar, S., Short, E.S., *et al.* (2018) Human Gaze Following for Human-Robot Interaction. 2018 *IEEE/RSJ International Conference on Intelligent Robots and Systems* (*IROS*), Madrid, 1-5 October 2018, 8615-8621. https://doi.org/10.1109/IROS.2018.8593580

[20] Vaswani, A., Shazeer, N., Parmar, N., *et al.* (2017) Attention Is All You Need. *Advances in Neural Information Processing Systems*, **30**, 5998-6008.

[21] 田永林, 王雨桐, 王建功, 等. 视觉 Transformer 研究的关键问题: 现状及展望[J]. 自动化学报, 2022, 48(4): 957-979.

[22] Dosovitskiy, A., Beyer, L., Kolesnikov, A., *et al.* (2020) An Image Is Worth 16x16 Words: Transformers for Image

Recognition at Scale.

[23] Carion, N., Massa, F., Synnaeve, G., *et al.* (2020) End-to-End Object Detection with Transformers. In: Vedaldi, A., Bischof, H., Brox, T. and Frahm, J.M., Eds., *Computer Vision—ECCV* 2020, *Lecture Notes in Computer Science*, Vol. 12346, Springer, Cham, 213-229. https://doi.org/10.1007/978-3-030-58452-8_13

[24] Cheng, Y. and Lu, F. (2022) Gaze Estimation Using Transformer. 2022 26*th International Conference on Pattern Recognition* (*ICPR*), Montreal, 21-25 August 2022, 3341-3347. https://doi.org/10.1109/ICPR56361.2022.9956687

[25] He, K., Zhang, X., Ren, S., *et al.* (2016) Deep Residual Learning for Image Recognition. *Proceedings of the IEEE Conference On Computer Vision and Pattern Recognition*, Las Vegas, 27-30 June 2016, 770-778. https://doi.org/10.1109/CVPR.2016.90

[26] Glorot, X. and Bengio, Y. (2010) Understanding the Difficulty of Training Deep Feedforward Neural Networks. *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, **9**, 249-256.

[27] Pan, J., Sayrol, E., Giro-i-Nieto, X., *et al.* (2016) Shallow and Deep Convolutional Networks for Saliency Prediction. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, 27-30 June 2016, 598-606. https://doi.org/10.1109/CVPR.2016.71