End-to-End Human-Gaze-Target Detection with Transformers

Danyang Tu[1], Xiongkuo Min[1], Huiyu Duan[1], Guodong Guo[2], Guangtao Zhai[1*], Wei Shen[3*]
[1]Institute of Image Communication and Network Engineering, Shanghai Jiao Tong University
[2]Institute of Deep Learning, Baidu Research, Beijing, China
[3]MoE Key Lab of Artificial Intelligence, AI Institute, Shanghai Jiao Tong University
{danyangtu, minxiongkuo, huiyuduan, zhaiguangtao, wei.shen}@sjtu.edu.cn, guoguodong01@baidu.com

Abstract

In this paper, we propose an effective and efficient method for Human-Gaze-Target (HGT) detection, i.e., gaze following. Current approaches decouple the HGT detection task into separate branches of salient object detection and human gaze prediction, employing a two-stage framework where human head locations must first be detected and then be fed into the next gaze target prediction sub-network. In contrast, we redefine the HGT detection task as detecting human head locations and their gaze targets, simultaneously. By this way, our method, named Human-Gaze-Target detection TRansformer or HGTTR, streamlines the HGT detection pipeline by eliminating all other additional components. HGTTR reasons about the relations of salient objects and human gaze from the global image context. Moreover, unlike existing two-stage methods that require human head locations as input and can predict only one human's gaze target at a time, HGTTR can directly predict the locations of all people and their gaze targets at one time in an end-to-end manner. The effectiveness and robustness of our proposed method are verified with extensive experiments on the two standard benchmark datasets, GazeFollowing and VideoAttentionTarget. Without bells and whistles, HGTTR outperforms existing state-of-the-art methods by large margins (6.4 mAP gain on GazeFollowing and 10.3 mAP gain on VideoAttentionTarget) with a much simpler architecture.

1. Introduction

Gaze following plays a crucial role in high level human-scene understanding tasks, and has attracted considerable research interest recently. Given an image or video frame containing one or more humans, the goal of gaze following is to predict where each person is looking at.

Unlike traditional gaze prediction tasks [11, 13] that predict only the gaze direction with a cropped human head image as input, gaze following further predicts the specific location in the scene that human is looking at. To this end, recent works leverage head pose features and the saliency maps of potential gaze targets by taking both head crops and the scene image as inputs. For instance, Recasens *et al*. [21] precisely detected the attention target of each person by extracting features from the scene and head images simultaneously. More recently, Chong *et al*. [4] proposed a novel framework to solve the problem of identifying gaze targets in videos. There are also other related works, including but not limited to [3, 7, 12, 13, 15, 30, 34]. These methods are very attractive since they demonstrate the ability to estimate gaze targets directly from images or videos, without the help of any monitor-based and wearable eye tracker devices.

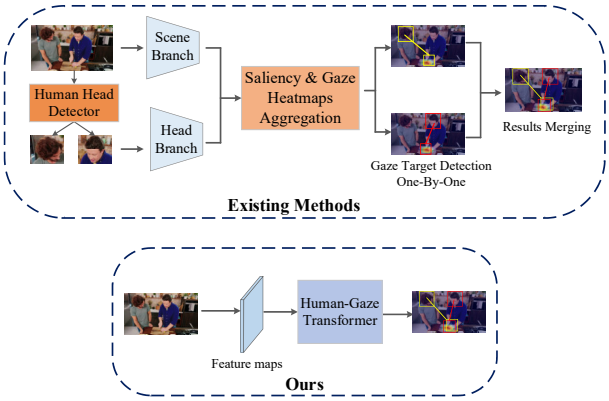

Figure 1. **Pipelines of existing methods and our proposed model**.

However, as shown in Figure 1, existing methods share a similar multi-stream architecture, which contains a scene branch for scene understanding and another parallel head branch to extract head pose features. In this case, a common problem arises in that both head images and scene images must be taken as inputs simultaneously. As a consequence, existing methods face the following three major drawbacks: (1) An additional human head detector is then essential in practical applications, which compels the entire framework

*Corresponding author

to be two-stage, and the precision of the additional detector can seriously affect the final results of gaze target detection. (2) As head crops are required for the second stage, existing methods can only predict gaze targets sequentially, which is less efficient when there are multiple persons in the same scene. It implies that the detection process will be conducted repeatedly and it is necessary to perform some post-operations to merge the detected gaze targets of different subjects in the same scene. (3) Most importantly, even if both head crops and scene images are taken as inputs, existing methods predict saliency maps and gaze directions separately, lacking contextual relational reasoning for interactions between them.

To overcome these drawbacks, we propose HGTTR, a Human-Gaze-Target detection TRansformer that simultaneously detects a human's head location and his/her attention target by image-wide contextual modeling. More specifically, taking as input the image of a scene containing one or more humans, our HGTTR is designed to simultaneously detect the head locations of all individuals as well as their gaze targets at one time. Its outputs can be represented as N human-gaze-target (HGT) instances in the format of $\langle$*head location, attention target*$\rangle$, where N is the number of persons in the image. Besides, with Transformers [27] as key component, HGTTR has a great ability to reason long-range gaze behaviors, thanks to its global contextual modeling capability.

To this end, we reformulate HGT detection as a set-based prediction problem. We define a HGT query set with several learnable embeddings, and each query is designed to capture at most one HGT instance. The HGTTR first takes a CNN backbone to extract high-level image features from only a single scene image, and then the encoder is leveraged to generate global memory features by modeling the relation between the image features explicitly. After that, the HGT queries and the global memory features are sent to the decoder to generate the output embeddings. Finally, the HGT instances are predicted based on the output embeddings of the decoder with a multi-layer perception. Meanwhile, we also propose a quaternate HGT matching loss to supervise the learning process of HGT instances prediction. Experimental results show that HGTTR outperforms existing state-of-the-art methods by large margins. Specifically, it achieves a 6.4 mAP gain on GazeFollowing [21] and a 10.3 mAP gain on VideoAttentionTarget [4] with a much higher FPS (more than 5 times compared to existing methods).

2. Related Work

Gaze following. Gaze following was first proposed in [21], which presented a large dataset, GazeFollowing, and an algorithm accordingly. Unlike eye tracking [13, 29, 32, 34] and saliency detection [12, 15], the goal of gaze following detection is to estimate what is being looked at by a person in a picture or a video frame. Based on this, Chong *et al.* [3] further addressed the problem of out-of-frame gaze targets by learning gaze angle and saliency simultaneously. By utilizing other auxiliary information, such as body pose [8], sight lines [16, 30], the within-frame gaze target estimation can be further enhanced. Besides, the work of [22] inferred gaze targets from videos. More recently, Chong *et al.* [4] proposed another new dataset named VideoAttentionTarget, which modeled the dynamics of gaze from video data and inferred per-frame gaze targets. In [7], a three-stage method was proposed to simulate the human gaze inference behavior in 3D space.

However, in both GazeFollowing and VideoAttentionTarget, human head locations were carefully and manually annotated and all existing methods took them as inputs, which is impossible for real world applications.

Transformer. Transformer was first proposed in Natural Language Processing (NLP) domain [27]. With self-attention mechanism as key component, it has great ability to selectively capture long range dependence among all tokens. Recently, a great number of excellent works have been proposed that perform Transformer structure in vision tasks. In DETR [1], object detection was reformulated as a set prediction problem and solved via a typical Transformer, which eliminates the need for many hand-designed components in object detection while demonstrating good performance. ViT [5] solved the image classification by representing an image as 16×16 patches and utilizing a Transformer encoder to predict the possible category. Transformer has also show great potential in many other vision tasks, such as semantic segmentation [24, 31], low-level vision tasks [2] and so on.

3. Method

3.1. Problem Reformulation

Given a single image $\boldsymbol{x} \in \mathbb{R}^{3\times H\times W}$ that contains one or more humans as input, we aim to predict all positions of all humans at one time, as well as their gaze locations. For convenience, we denote the position of people $\boldsymbol{l}_h$ as their head locations since they have been annotated in existing datasets. Specifically, a human's head location is represented as a bounding box $(x_{tl}, y_{tl}, x_{br}, y_{br})$, where the subscripts tl and br denote top-left and bottom-right, respectively. Meanwhile, the gaze location $\boldsymbol{l}_g$ is represented as a Gaussian heatmap. By this means, our problem can be formulated as maximizing a joint *posteriori* of the output pairs $\langle \boldsymbol{l}_h, \boldsymbol{l}_g \rangle$ on the input image:

$$\mathcal{T}^* \doteq \max_{\mathcal{T}} \prod_{i=1}^{N} p(\langle \boldsymbol{l}_h, \boldsymbol{l}_g \rangle_i \mid \boldsymbol{x}), \tag{1}$$

where N is the number of the humans in this image and $\mathcal{T}^*$ refers to the optimal model.

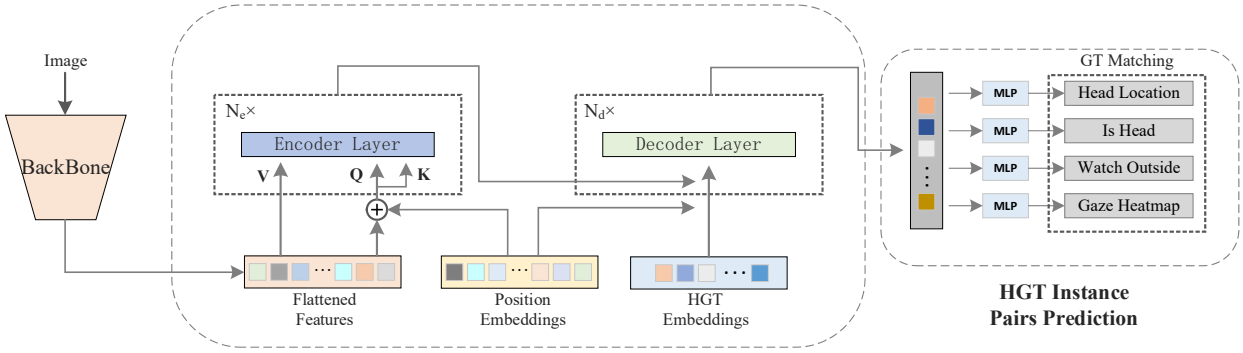

Figure 2. **Pipeline of the proposed model**. It consists of four key components: a backbone, a typical Transformer, four multi-layer perceptions (MLP) and a quaternate loss function.

It has to be noticed that this is quite different from the traditional gaze following problem, which takes both scene image and cropped human head image as input, and predicts only one individual's gaze location at a time. It can be formulated as:

$$\mathcal{T}^* \doteq \max_{\mathcal{T}} p(\boldsymbol{l}_g^j \mid \boldsymbol{x}, \boldsymbol{x}_h^j), \tag{2}$$

where $\boldsymbol{x}_h^j$ refers to the j-th cropped human head image and $\boldsymbol{l}_g^j$ denotes the j-th human's gaze location. Namely, for existing methods, an identical scene image is regarded as N different cases when there are N different humans.

3.2. Network Architecture

Figure 2 illustrates the overall architecture of the proposed HGTTR. It consists of four main parts: (i) a backbone to extract high-level visual feature from the input image, (ii) a Transformer encoder-decoder to digest visual feature and generate output embeddings, (iii) several multi-layer perceptions (MLP) to predict HGT instances and (iiii) a quaternate loss function for bipartite matching.

Backbone. Given an input image $\boldsymbol{x} \in \mathbb{R}^{3\times H\times W}$, a feature map $\boldsymbol{z_b} \in \mathbb{R}^{D_b\times H'\times W'}$ is calculated by an arbitrary CNN backbone network. $\boldsymbol{z_b}$ is then fed into a projection convolution layer with a kernel size of 1×1 to reduce the dimension from D_b to D_c. After that, a flatten operator is used to collapse the spatial dimension into one dimension, and a feature map $\boldsymbol{z}_f \in \mathbb{R}^{D_c\times HW}$ is obtained, which is denoted as *Flattened Features* in Figure 2. In this work, we use ResNet [9] as our backbone and reduce the dimension of feature layer-5 from $D_b = 2048$ to $D_c = 256$.

Encoder. The encoder consists of N_e encoder layers built upon standard Transformer structure with a multi-head self-attention (MSA) module and a feed-forward network (FFN). It takes the output of the backbone $\boldsymbol{z}_f \in \mathbb{R}^{D_c\times HW}$ as input to produce another feature map with richer contextual information. Besides, a fixed positional encoding $\boldsymbol{p} \in \mathbb{R}^{D_c\times H'\times W'}$ is additionally fed into the encoder to supplement the positional information as the Transformer architecture is permutation invariant. The attention in Transformer can be formulated as:

$$\begin{aligned} \text{Attention}(Q,K,V) &= \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V, \\ Q = Q_f + Q_p, K &= K_f + K_p, V = V_f, \end{aligned} \tag{3}$$

where d_k is the channel dimension, subscript f means the feature and p refers to the position encoding. The Q, K, V are the query, key and value, respectively. As shown in Figure 2, in the self-attention module of the encoder, $Q_f = K_f = V_f = \boldsymbol{z}_f$, and $Q_p = K_p = \boldsymbol{p}$.

Decoder. We also build the decoder layer on the basis of transformer architecture. Different from the encoder layer containing only self-attention module, the decoder layer consists of both self-attention and cross-attention mechanisms. In self-attention, K_f and V_f are the same as Q_f while K_p is the same as Q_p. Specifically, $Q_f \in \mathbb{R}^{N_q\times C}$ is the output of the last decoder and we initialize it for the first decoder with a constant vector. For HGT query position embedding $Q_p \in \mathbb{R}^{N_q\times C}$, we use a set of learning embedding:

$$Q_p = \text{Embedding}(N_q, C), \tag{4}$$

where N_q is always larger than the number of actual HGT instances in an image. In cross-attention, $Q_f \in \mathbb{R}^{N_q\times C}$ is generated from the output of the former self-attention module while $K_f \in \mathbb{R}^{HW\times C}$ and $V_f \in \mathbb{R}^{HW\times C}$ are the output features of the encoder. Q_p is also set as Eq. (4) and K_p is equal to $\boldsymbol{p}$.

In summary, the decoder has three inputs, the global memory from the encoder, HGT instance queries and position encoding. It serves to transform N_q learnable position embeddings (denoted as *HGT instance Embeddings* in Figure 2) into N_q output embeddings with both the self-attention and cross-attention mechanisms.

MLP for HGT instances prediction. Mathematically, we define each HGT instance by the following three vectors: a human-head-bounding-box vector normalized by the corresponding image size $\boldsymbol{l}_h \in [0,1]^4$, a watch-in-out (whether the gaze target is located inside the scene image or not) binary one-hot vector $\boldsymbol{w} \in \{0,1\}^2$, and a gaze heatmap vec-

tor $\boldsymbol{l}_g \in [0,1]^{H_o \times W_o}$, where H_o and W_o denote the spatial resolution of the output gaze heatmap.

The output embedding for each HGT query is decoded to one HGT instance by several multi-layer perception (MLP) branches. Specifically, we use two one-layer MLP branches f_h and f_o to predict the human confidence (whether the predicted bounding box is a human or not) and watch-in-out confidence, respectively. Meanwhile, a three-layer MLP branch f_{lh} is set to predict human-head-bounding-box as well as a five-layer MLP branch f_{lg} to predict the gaze heatmap. We use a softmax function for all one-layer branches and a sigmoid function for box and heatmap prediction.

3.3. Loss Calculation

The loss calculation is composed of two stages: the bipartite matching between ground-truths and output predictions of the proposed network, and the loss calculation for the matched pairs.

Bipartite matching. We follow the training procedure of DETR [1] and use the Hungarian algorithm [14] for the bipartite matching, which is designed to obviate the process of suppressing over-detection. First of all, we pad the ground-truth of HGT instances with $\varnothing$ (no instance) so that the size of ground-truths set becomes N_q.

As illustrated in Figure 2, the model outputs a fixed-size set of N_q HGT instance predictions, and we denote them as $O = o^i, i = 1, 2, \cdots, N_q$. Meanwhile, we use $T = t^i, i = 1, 2, \cdots M, \varnothing_1, \varnothing_2, \cdots, \varnothing_{N_q - M}$ to represent the ground-truths, where M is the real number of HGT instances in an image. Then, the matching process can be denoted as an injective function: $\omega_{T \to O}$, where $\omega(i)$ is the index of predicted HGT instance assigned to the i-th ground-truth. We define the matching cost as:

$$\mathcal{L}_{cost} = \sum_{i}^{N_q} \mathcal{L}_{match}(t^i, o^{\omega(i)}), \tag{5}$$

where $\mathcal{L}_{match}(t^i, o^{\omega(i)})$ is a matching cost between the i-th ground-truth and the $\omega(i)$-th prediction.

Specifically, the matching cost $\mathcal{L}_{match}(t^i, o^{\omega(i)})$ consists of four types of cost: the head-box-regression cost $\mathcal{L}_{box}$, is-head cost $\mathcal{L}_h$, watch-in-out cost $\mathcal{L}_w$ and gaze heatmap cost $\mathcal{L}_g$. $\mathcal{L}_{box}$ is box regression loss for human head box, and the weighted sum of GIoU [23] loss and L_1 loss is used:

$$\mathcal{L}_{box} = \alpha_1 \left\| t_i^b - o_{\omega(i)}^b \right\| - \alpha_2 \text{GIoU}(t_i^b, o_{\omega(i)}^b), \tag{6}$$

where the superscript b refers to the bounding box. Besides, the $\mathcal{L}_h$ and $\mathcal{L}_w$ are respectively defined as:

$$\mathcal{L}_h = -o_{\omega(i)}^c(k) \;\; s.t. \;\; t_i^c(k) = 1, \tag{7}$$

$$\mathcal{L}_w = -o_{\omega(i)}^w(k) \;\; s.t. \;\; t_i^w(k) = 1, \tag{8}$$

where $c \in \{0,1\}^2$ and $w \in \{0,1\}^2$ are one-hot vector for is-head and watch-in-out, respectively. We use L_2 loss for heatmap cost $\mathcal{L}_g$:

$$\mathcal{L}_g = \left\| t_i^g - o_{\omega(i)}^g \right\|_2. \tag{9}$$

On this basis, we design the following matching cost for HGT prediction:

$$\mathcal{L}_{match}(t^i, o^{\omega(i)}) = \beta_1 \mathcal{L}_{box} + \beta_2 \mathcal{L}_h + \beta_3 \mathcal{L}_w + \beta_4 \mathcal{L}_g. \tag{10}$$

We then leverage the Hungarian algorithm to determine the optimal assignment $\hat{\omega}$ among the set of all possible permutations of N_q elements $\boldsymbol{\Omega}_{N_q}$. It can be formulated as:

$$\hat{\omega} = \arg\min_{\omega \in \boldsymbol{\Omega}_{N_q}} \mathcal{L}_{cost}. \tag{11}$$

Loss function. After the optimal one-to-one matching between the ground-truths and the predictions is found, the loss to be minimized in the training phase is calculated as Eq. (10). The hyper-parameters are set as same as they does in matching process.

4. Experiments

4.1. Datasets and Evaluation Metrics

Datasets. We train and test our model on both GazeFollowing [21] dataset and VideoAttentionTarget [4] dataset. Specifically, we use every single frame in VideoAttentionTarget dataset as input during the training process, without considering the temporal information. Therefore, to avoid overfitting, for every five continuous frames from the training set of VideoAttentionTarget which have no obvious appearance differences, we randomly select one for training since they have almost the same gaze target. For testing, we still use all images in the testing set.

Moreover, one of objectives of our proposed model is to predict the locations of different individuals. Therefore, the head locations in existing dataset annotations are no longer used as inputs but as ground-truths. Besides, previous works predict the gaze target for different subjects in an identical scene on a case-by-case basis, which results in each image being assigned with M annotations, and M is the number of people in an image. Unlike that, we merge the annotations of the same image into the same format as COCO [17] object detection since we aim to predict them all at one time.

Evaluation metric. In this work, we have to evaluate the performance of proposed model in terms of both gaze target detection and human position detection.

For the former, we follow the standard evaluation protocols, as in [4, 21], to report the results in terms of **AUC** and $\boldsymbol{L_2}$ distance. AUC: The final heatmap provides the prediction confidence score which is evaluated at different thresholds in the ROC curve. The area under curve (AUC) of the ROC is reported [4]. Distance: L_2 distance between the annotated target location and the prediction given by the pixel having the maximum value in the heatmap, with image width and height normalized to 1. Specifically, since the

Method	GazeFollowing							VideoAttentionTarget						
	AUC↑		Average Dist.↓		Min Dist.↓		mAP↑	AUC↑		L2 Dist.↓		AP↑		mAP↑
	Default	**Real**	Default	**Real**	Default	**Real**		Default	**Real**	Default	**Real**	Default	**Real**	
Random	0.504	0.391	0.484	0.533	0.391	0.487	0.104	0.505	0.247	0.458	0.592	0.621	0.349	0.091
Center	0.633	0.446	0.313	0.495	0.230	0.371	0.117	—	—	—	—	—	—	—
Fixed bias	—	—	—	—	—	—	—	0.728	0.522	0.326	0.472	0.624	0.510	0.130
Judd [12]	0.711	—	0.337	—	0.250	—	—	—	—	—	—	—	—	—
GazeFollow [21]	0.878	0.804	0.190	0.233	0.113	0.124	0.457	—	—	—	—	—	—	—
Chong [3]	0.896	0.807	0.187	0.207	0.112	0.120	0.449	0.830	0.791	0.193	0.214	0.705	0.651	0.374
Zhao [30]	—	—	0.147	—	0.082	—	—	—	—	—	—	—	—	—
Lian [16]	0.906	0.881	0.145	0.153	0.081	0.087	0.469	0.837	0.784	0.165	0.172	—	—	0.392
VideoAttention [4]	0.921	0.902	0.137	0.142	0.077	0.082	0.483	0.860	0.812	0.134	0.146	0.853	0.849	0.420
DAM [7]	0.922	—	0.124	—	0.067	—	—	0.905	—	0.108	—	0.896	—	—
HGTTR (ResNet-50)	—	**0.917**	—	**0.133**	—	0.069	**0.547**	—	0.893	—	0.137	—	0.821	0.514
HGTTR (ResNet-101)	—	0.905	—	0.138	—	**0.065**	0.541	—	**0.904**	—	**0.126**	—	**0.854**	**0.523**

Table 1. **Quantitative comparisons on the GazeFollowing and VideoAttentionTarget sets.** As the existing methods require to take as input both scene and head images, we report the results of them in 'Default' and 'Real' set, respectively. Specifically, 'Default' refers to use the head location that carefully labeled in existing dataset, while 'Real' represents to apply an additional head detection network to predict the head location for real world applications.

ground truth for GazeFollowing may be multimodal, the L_2 distance is the Euclidean distance between our prediction and the average of ground-truth annotations. Besides, the minimum distance between our prediction and all ground-truth annotations is also reported. In addition, the average precision (**AP**) is used to evaluate the performance for is-watching-outside prediction.

For the latter, we use the commonly used role *mean average percision* (**mAP**) to examine the model performance on both datasets. Specifically, a HGT detection is considered as true positive if and only if it localizes the human and detects gaze target accurately (*i.e.* the *Interaction-over-Union* (IOU) ratio between the predicted human-head-box and ground-truth is greater than 0.5 while the L_2 distance for gaze target detection is less than 0.15).

4.2. Implementation Details

The experiments are conducted on two popular backbone: ResNet-50 and ResNet-101 [9]. Both Transformer encoder and decoder consist of 6 layers with a multi-head self-attention of 8 heads. We initialize the network with the parameters of DETR trained with the COCO dataset. The model is trained for 150 epochs using AdamW [19] optimizer with batch size of 16. Specifically, the initial learning rate of the backbone network is set as 10^{-5} while that of the others is set as 10^{-4}, the weight decay is equal to 10^{-4}. For the hyper-parameters of the model, we set α_1 and α_2 as 1.0 and 2.5, respectively. The weights $(\beta_1 - \beta_4)$ for different cost functions in the loss are set as 2, 1, 1, 2, respectively. The number of HGT instance queries N_q is 20 for all datasets. Both learning rates are decayed after 80 epochs. All experiments are conducted on 8 NVIDIA GTX 2080TI GPU.

4.3. Comparison to State-of-the-Art

We first show the main quantitative comparison of our HGTTR with the latest HGT detection methods in Table 1. Specifically, since the cropped head image and the head location are essential for existing methods, we report the results of them on two different settings: 'Default' and 'Real'. For 'Default', we directly utilize the carefully labeled head locations in existing dataset annotations. Meanwhile, we employ an additional head detector to automatically generate the head positions and feed the predicted results into existing models for real world applications, and the results are reported as 'Real'. Following [4], we fine-tuned a SSD-based [18] head detection network with the head annotations in existing dataset. As can be seen from the table, HGTTR outperforms existing state-of-the-art methods on all datasets. HGTTR with the ResNet-50 backbone yields a significant gain of 6.4 mAP compared with VideoAttention [4] and 7.8 mAP compared with GazeFollow [21] on the GazeFollowing datasets. Moreover, HGTTR performs better on the VideoAttentionTarget dataset. This can mainly attribute to two main reasons: 1) VideoAttentionTarget is a larger dataset than GazeFollowing, which is important for transformer-based methods. 2) There are more than one humans in each image of VideoAttentionTarget while every image in GazeFollowing contains only one person. As our model outputs a fixed number of HGT instances, fewer instances in an image imply a higher value of fault positive (FP). In terms of gaze target prediction, our HGTTR still has outstanding performance. For VideoAttentionTarget dataset, we achieve a gain of 8 AUC compared with VideoAttention for 'Real' setting.

4.4. Ablation Study

In this section, we conduct extensive experiments to validate the effectiveness of our proposed HGTTR. The abla-

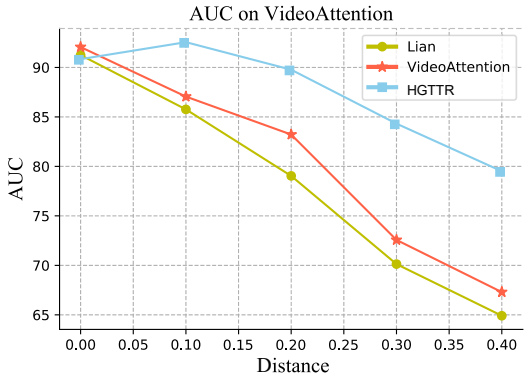

Figure 3. **Performance of different methods on different spatial distributions of HGT instances on VideoAttentionTarget.**

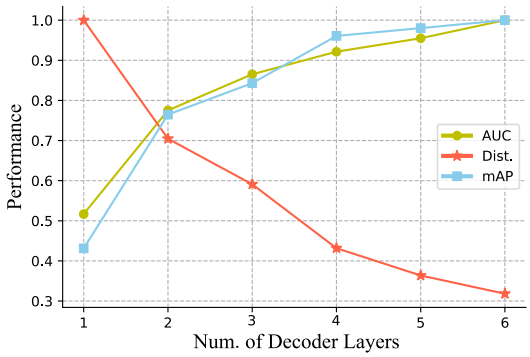

Figure 4. **Performance of the model with different numbers of decoder layers.** To show them in the same figure, all metrics are normalized by dividing by the maximum value. Namely, we mainly show their trends as the number of decoder layers increases.

tion experiments are conducted with ResNet-50 backbone model, and the results are reported on the VideoAttentionTarget dataset.

4.4.1 Model components

Self-attention. As the most important component in transformer, self-attention has great ability to capture long range contextual information. In [4], attention mechanism has been simply applied to merge the features of head branch and scene branch, which achieved attractive performance. We first investigate in which cases self-attention especially achieves superior performance compared with the existing methods. To this end, we split HGT instances into bins of size 0.1 on the basis of L_1 distance between the center of a human's head and his gaze target, where the height and width of image have been normalized. The AUC of different methods in each bin is shown in Figure 3, where the results of Lian [16] and VideoAttention [4] are reported in 'Default' setting. The relative gaps of the performance between HGTTR and the other two methods become more evident as the distance grows. It indicts that gaze target detection tends to become more difficult as the distance grows. Besides, it is especially difficult for existing methods to deal with the distant cases while HGTTR has relatively better performance. The possible reason for such result is that existing methods rely on limited receptive fields for the feature aggregation. They are weak in capturing long range contextual information or easily be dominated by irrelevant information in the distant cases. On the contrary, the features of HGTTR are more effective thanks to the ability of self-attention to adaptively extract image-wide contextual information.

Decoder. A decoder is essential to transform the manually defined HGT queries Q_f into a set of HGT instances with the features generated by the encoder. As each layer of the decoder is identical in the architecture, it is a coarse-to-fine process where each layer takes the predicted results of the previous layer as input to further produce more precise predictions. As shown in Figure 4, HGTTR can achieve better performance with more decoder layers, especially for 'AUC' and 'L_2 Dist.'. A possible explain for this result is that the decoder can learn some potential relationships among different HGT instances with more self-attention mechanisms. For example, people in an identical scene more tend to have a similar gaze target.

β_1	β_2	β_3	β_4	AUC↑	L_2 Dist.↓	mAP↑
1	1	1	1	0.864	0.142	0.492
1	2	2	1	0.857	0.148	0.487
2	1	1	2	**0.893**	**0.137**	**0.514**

Table 2. **The effects of different cost functions in matching process**.

Moreover, we visualize the decoder attention map for the predicted HGT instances in Figure 5 . The heatmap highlights both the human heads and their gaze targets, which indicts that our HGTTR reasons about the relations between human and scene from a more global image context.

Matching strategy. Our proposed matching cost function consists of four main aspects: human head location $\mathcal{L}_{box}$, whether a predicted bounding box is a human or not (Is Head) $\mathcal{L}_h$, whether the gaze target is located in the scene image or not (Watch Outside) $\mathcal{L}_w$ and the predicted gaze heatmap $\mathcal{L}_g$. Specifically, $\mathcal{L}_{box}$ and $\mathcal{L}_g$ are localization losses while $\mathcal{L}_h$ and $\mathcal{L}_w$ are classification losses. In this case, we conduct ablation study to further find the relative importance in matching: location first or classification first? In Eq. (10), β_1 to β_4 denote the different weights for each matching cost function. As shown in Table 2, the best result is obtained under $\beta_1 = \beta_4 = 2$ and $\beta_2 = \beta_3 = 1$, which suggests that localization plays a relatively more important role than classification during the matching process.

4.4.2 Importance of pairwise detection

We redefine the gaze following task to detect the human head locations and their gaze targets, simultaneously. In this

Figure 5. **Visualization of attention maps in the decoder for the predicted HGT instances.** The images are randomly selected from the testing set of VideoAttentionTarget. As can be seen from the figure, our method has great ability to capture long range dependence. In addition, it can be seen from the figure that different HGT instances may share little common pattern (the last column), which indicts that an unique matching process is essential.

Method	Fine-tuned	FPS	Blur		Gaussian Noise		Brightness		Normal	
			AUC	mAP	AUC	mAP	AUC	mAP	AUC	mAP
SSD-based [18]	✗	2	0.729	0.302	0.706	0.298	0.738	0.315	0.789	0.405
	✓	2	0.774	0.371	0.767	0.362	0.784	0.376	0.812	0.420
FCHD [28]	✗	3	0.757	0.336	0.744	0.328	0.764	0.341	0.804	0.409
	✓	3	0.784	0.385	0.781	0.325	0.796	0.392	0.818	0.424
HeadHunter [26]	✗	1	0.782	0.341	0.750	0.332	0.773	0.352	0.796	0.412
	✓	1	0.787	0.389	0.789	0.384	0.804	0.410	0.819	0.424
HGTTR (Ours)	✗	16	0.806	0.442	0.792	0.438	0.813	0.451	0.893	0.514
	✓	**16**	**0.872**	**0.487**	**0.864**	**0.481**	**0.881**	**0.496**	—	—

Table 3. **Performance of different human head detectors**. We manually degrade the original images in the testing set with several common distortion types. Specifically, 'Fine-tuned' refers to that the training set is also augmented with these distortion types and the pre-trained head detector is further fine-tuned with the head locations in annotations. 'Normal' denotes that no degeneration image is used.

subsection, we mainly analyze the necessity of this strategy.

Model robustness. As existing methods take both scene and human head images as inputs, an additional human head detector is essential for them. However, performance of gaze target prediction would then be seriously influenced by the precision of the head detector in this way. As shown in Table 3, we applied several different head detectors to analyze model performance in different conditions. In real world applications, such as video surveillance, blur, noise, and brightness variation are the most common types of image degradation. It can be seen from the table, pairwise detection strategy is not only more efficient, but also more robust with better performance under different degradations. First, using an additional head detector is a sub-optimal solution since existing models are trained with head locations that are carefully and manually labeled so that they are very sensitive to the results of the head detector. Secondly, while noise can be addressed to some extent by data augmentation in real world applications, head detector is hard to fine-tune since it is impossible to manually generate annotations of head locations. On the contrary, we solve HGT detection in an absolute end-to-end manner, which does not require head location as input. Moreover, as shown in the table, HGTTR has better robustness with different image degradations. The possible reasons are that Transformer inherently has great ability to against noises and pairwise detection is not as sensitive to image degradations as two-stage methods.

Model efficiency. Another advantage of pairwise detection is high efficiency. Using human head images as input, as existing methods do, limits the ability of the model to predict different people's gaze targets at the same time. As shown in Figure 6, as the number of individuals in an identical scene increases, the inference time of existing methods grows extremely. The main reason is that existing methods can predict only one human's gaze target at a time, and the detection process has to be conducted repeatedly when there are more than one human in a same image. On the contrary, our proposed method has no such limitation and the inference time is almost not inflected by the number of humans.

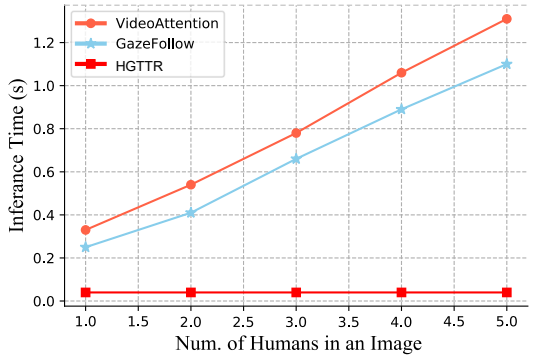

Figure 6. **Inference time of different models with different numbers of humans in an image.**

In this paper, our HGTTR is designed to detect up to 20 peoples gaze targets at a time. In real world applications, the model can be easily fine-tuned to increase the number of maximum detectable humans by simply changing the value of hyper-parameter N_q.

4.5. Qualitative Analysis

Practical application. To evaluate our model's performance in practical applications, we randomly selected 15,000 pictures from the DL Gaze dataset [16], which records the daily activities of 16 volunteers in 4 different scenes, and the ground-truth is annotated by the observers in the videos. All models are only trained with the VideoAttentionTarget dataset. In addition, an additional human head detector is also employed for existing methods. The quantitative results are presented in Table 4. Without bells and whistles, our HGTTR outperforms all state-of-the-art methods by a significant margin.

Shared attention detection. With the ability to detect gaze targets of different people at a same time, our methods are inherently suitable for inferring shared attention in social scenes. Following [4], the results are reported in terms of accuracy for interval detection of shared attention and L_2 distance for location prediction on the VideoCoAtt [6] dataset. This dataset has 113,810 test frames that are annotated with the target location when it is simultaneously attended by two or more people. As can be seen from Table 5, our model has demonstrated potential value of recognizing higher-level social gaze.

4.6. Discussion

It is our belief that defining HGT detection task as simultaneously detecting human head locations and their gaze targets is more reasonable, since manually labeled head locations are unavailable in practical applications. Moreover, joint detection could benefit both of these two tasks with the essential contextual information modeling. However, as a typical Transformer-based method, HGTTR suffers from slow convergence, since it takes long training epochs to learn attention weights to focus on sparse meaningful locations. A possible solution to this issue is designing a more flexible self-attention mechanism with more considerations on this task, like deformable DETR [33]. We will devote more studies on this in the future work.

Methods	AUC↑	L_2 Dist.↓	Ang.↓	mAP↑
Gazefollow [21]	0.792	0.213	27.9°	0.407
Lian [16]	0.813	0.167	19.7°	0.441
VideoAttention [4]	0.842	0.145	16.9°	0.476
HGTTR	**0.912**	**0.121**	**12.7°**	**0.538**

Table 4. **Performance in practical application**. Specifically, 'Ang.' refers to the angle between ground-truth gaze direction and the predicted one.

Methods	Accuracy↑	L_2 Dist.↓
Random	50.8	286
Fixed bias	52.4	122
GazeFollow [21]	58.7	102
Gaze+Saliency [20]	59.4	83
Gaze+Saliency+LSTM [10]	66.2	71
Fan [6]	71.4	62
Sumer [25]	78.1	63
VideoAttention [4]	83.3	57
HGTTR	**90.4**	**46**

Table 5. **Shared attention detection results on the VideoCoAtt dataset**. The interval detection is evaluated in terms of prediction accuracy while the location task is measured with L_2 distance.

Broader impacts. The proposed method predicts human's gaze target. As a human-centric task, it may has some issues about privacy protection when being applied practically, which warrants more policy reviews when using this work in real world applications.

5. Conclusion

We have presented an new Transformer-based method for the task of gaze following detection. Our model is designed to allow detect all people's head positions as well as their gaze targets simultaneously. Without the limitation of taking head image as input, our model achieve higher effectiveness and efficiency. Extensive experiments validate its strong performance as well as the potential for understanding gaze behavior in naturalistic human interactions. We hope our method will be useful for human activity understanding research.

Acknowledgments. This work was supported by NSFC 61831015, National Key RD Program of China 2021YFE0206700, NSFC 62176159, Natural Science Foundation of Shanghai 21ZR1432200 and Shanghai Municipal Science and Technology Major Project 2021SHZDZX0102.

References

[1] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, pages 213–229, 2020.

[2] Hanting Chen, Yunhe Wang, Tianyu Guo, Chang Xu, Yiping Deng, Zhenhua Liu, Siwei Ma, Chunjing Xu, Chao Xu, and Wen Gao. Pre-trained image processing transformer. In *CVPR*, pages 12299–12310, 2021.

[3] Eunji Chong, Nataniel Ruiz, Yongxin Wang, Yun Zhang, Agata Rozga, and James M Rehg. Connecting gaze, scene, and attention: Generalized attention estimation via joint modeling of gaze and scene saliency. In *ECCV*, pages 383–398, 2018.

[4] Eunji Chong, Yongxin Wang, Nataniel Ruiz, and James M Rehg. Detecting attended visual targets in video. In *CVPR*, pages 5396–5406, 2020.

[5] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021.

[6] Lifeng Fan, Yixin Chen, Ping Wei, Wenguan Wang, and Song-Chun Zhu. Inferring shared attention in social scene videos. In *CVPR*, pages 6460–6468, 2018.

[7] Yi Fang, Jiapeng Tang, Wang Shen, Wei Shen, Xiao Gu, Li Song, and Guangtao Zhai. Dual attention guided gaze target detection in the wild. In *CVPR*, pages 11390–11399, 2021.

[8] Jian Guan, Liming Yin, Jianguo Sun, Shuhan Qi, Xuan Wang, and Qing Liao. Enhanced gaze following via object detection and human pose estimation. In *ICMM*, pages 502–513, 2020.

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

[10] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, pages 1735–1780, 1997.

[11] Qiong Huang, Ashok Veeraraghavan, and Ashutosh Sabharwal. Tabletgaze: dataset and analysis for unconstrained appearance-based gaze estimation in mobile tablets. *Machine Vision and Applications*, pages 445–461, 2017.

[12] Tilke Judd, Krista Ehinger, Frédo Durand, and Antonio Torralba. Learning to predict where humans look. In *ICCV*, pages 2106–2113, 2009.

[13] Kyle Krafka, Aditya Khosla, Petr Kellnhofer, Harini Kannan, Suchendra Bhandarkar, Wojciech Matusik, and Antonio Torralba. Eye tracking for everyone. In *CVPR*, pages 2176–2184, 2016.

[14] Harold W Kuhn. The hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, pages 83–97, 1955.

[15] George Leifman, Dmitry Rudoy, Tristan Swedish, Eduardo Bayro-Corrochano, and Ramesh Raskar. Learning gaze transitions from depth to improve video saliency estimation. In *CVPR*, pages 1698–1707, 2017.

[16] Dongze Lian, Zehao Yu, and Shenghua Gao. Believe it or not, we know what you are looking at! In *ACCV*, pages 35–50, 2018.

[17] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, pages 740–755, 2014.

[18] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *ECCV*, pages 21–37, 2016.

[19] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. 2017.

[20] Junting Pan, Elisa Sayrol, Xavier Giro-i Nieto, Kevin McGuinness, and Noel E O'Connor. Shallow and deep convolutional networks for saliency prediction. In *CVPR*, pages 598–606, 2016.

[21] Adria Recasens, Aditya Khosla, Carl Vondrick, and Antonio Torralba. Where are they looking? In *NIPS*, pages 199–207, 2015.

[22] Adria Recasens, Carl Vondrick, Aditya Khosla, and Antonio Torralba. Following gaze in video. In *CVPR*, pages 1435–1443, 2017.

[23] Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. Generalized intersection over union: A metric and a loss for bounding box regression. In *CVPR*, pages 658–666, 2019.

[24] Robin Strudel, Ricardo Garcia, Ivan Laptev, and Cordelia Schmid. Segmenter: Transformer for semantic segmentation. *arXiv preprint arXiv:2105.05633*, 2021.

[25] Omer Sumer, Peter Gerjets, Ulrich Trautwein, and Enkelejda Kasneci. Attention flow: End-to-end joint attention estimation. In *CVPR*, pages 3327–3336, 2020.

[26] Ramana Sundararaman, Cedric De Almeida Braga, Eric Marchand, and Julien Pettre. Tracking pedestrian heads in dense crowd. In *CVPR*, 2021.

[27] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, pages 5998–6008, 2017.

[28] Aditya Vora and Vinay Chilaka. Fchd: Fast and accurate head detection in crowded scenes. *arXiv preprint arXiv:1809.08766*, 2018.

[29] Xucong Zhang, Yusuke Sugano, Mario Fritz, and Andreas Bulling. Appearance-based gaze estimation in the wild. In *CVPR*, pages 4511–4520, 2015.

[30] Hao Zhao, Ming Lu, Anbang Yao, Yurong Chen, and Li Zhang. Learning to draw sight lines. *IJCV*, pages 1–25, 2019.

[31] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip HS Torr, et al. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In *CVPR*, pages 6881–6890, 2021.

[32] Wangjiang Zhu and Haoping Deng. Monocular free-head 3d gaze tracking with deep learning and geometry constraints. In *CVPR*, pages 3143–3152, 2017.

[33] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. In *ICLR*, 2020.

[34] Zhiwei Zhu and Qiang Ji. Eye gaze tracking under natural head movements. In *CVPR*, pages 918–923, 2005.