# MGTR: End-to-End Mutual Gaze Detection with Transformer

Hang Guo, Zhengxi Hu, and Jingtai Liu [⋆]

Nankai University, Tianjin, China
{1911610, hzx}@mail.nankai.edu.cn, liujt@nankai.edu.cn

**Abstract.** People's looking at each other or mutual gaze is ubiquitous in our daily interactions, and detecting mutual gaze is of great significance for understanding human social scenes. Current mutual gaze detection methods focus on two-stage methods, whose inference speed is limited by the two-stage pipeline and the performance in the second stage is affected by the first one. In this paper, we propose a novel one-stage mutual gaze detection framework called Mutual Gaze TRansformer or MGTR to perform mutual gaze detection in an end-to-end manner. By designing mutual gaze instance triples, MGTR can detect each human head bounding box and simultaneously infer mutual gaze relationship based on global image information, which streamlines the whole process with simplicity. Experimental results on two mutual gaze datasets show that our method is able to accelerate mutual gaze detection process without losing performance. Ablation study shows that different components of MGTR can capture different levels of semantic information in images. Code is available at https://github.com/Gmbition/MGTR

**Keywords:** End-to-End Mutual Gaze Detection · One-stage Method · Mutual Gaze Instance Match

## 1 Introduction

Containing rich information, the gaze plays an important role in reflecting the attention, intention and emotion of one person [2,1]. Among all kinds of gaze, mutual gaze is indispensable in building the bridge between two minds [3,17]. From mutual gaze, one can infer the willingness to interact and the strength of the relationship. Moreover, mutual gaze can also be used for people connection analysis in social scene interpretation and is an important clue for Human-Robot-Interaction. For these reasons, it is very promising to achieve automatic mutual gaze detection.

The target for end-to-end mutual gaze detection is to detect all the human heads in the scene and then recognize whether any two people are looking at each

This work is supported in part by National Key Research and Development Project under Grant 2019YFB1310604, in part by National Natural Science Foundation of China under Grant 62173189.

[⋆] Corresponding author

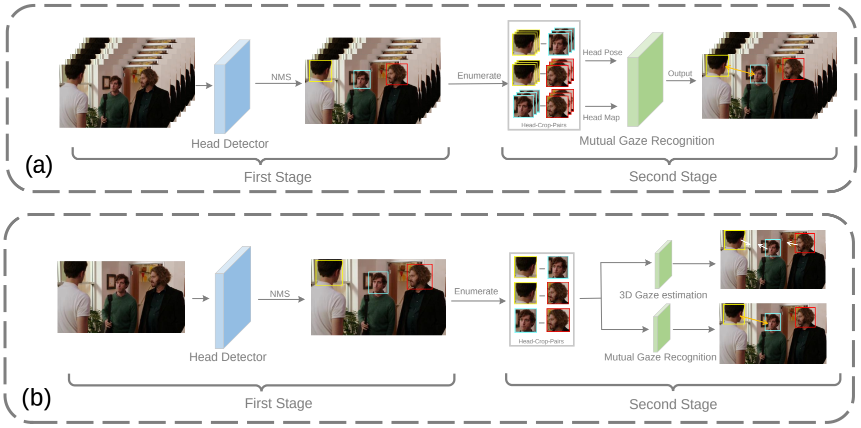


**Fig. 1.** An illustration of two-stage methods. The method in (a) uses a pretrained head detector and performs mutual gaze recognition by exploiting pose and position information of enumerated head-crop-pairs. The method in (b) also uses a head detector and utilizes pseudo 3D gaze to boost mutual gaze detection. It can be seen that methods in (a) and (b) both perform head detection in the first stage and need to enumerate all head-crop-pairs in one image which slows down the inference process.

other. Previous studies [20,6,19] have got favorable results by dividing the whole process into two stages: detect all human heads in the scene and then regard the recognition of mutual gaze as a binary classification problem. Specifically, Marin *et al* proposed a video based method [19] (Fig.1(a)) that first detects all human heads through a pretrained head detector and then enumerates all head-crop-pairs and put them into a classification network to identify whether two people are looking at each other. The image based mutual gaze detection work in Doosti *et al* [6] (Fig.1(b)) uses pseudo 3D gaze to boost mutual gaze detection and also adapts a two-stage strategy.

Although these two-stage methods have achieved promising results, their designs have some shortcomings. Firstly, the classifier in the second stage makes inferences based on the local information of the head crop instead of the image global information. For example, body posture is also an important clue for judging mutual gaze. Moreover, the performance of the classification results in the second stage depends on the localization accuracy of the first stage. Furthermore, when there are many people in the scene, the computational cost will also increase due to the need to enumerate all detected heads which will slow down the inference process.

To overcome these limitations, inspired by the attention mechanism in Transformer Network [25], we propose a *one-stage* mutual gaze detection model called Mutual Gaze TRansformer or MGTR (Fig.2) which can detect all human heads in the scene and simultaneously identify whether there is a mutual gaze based

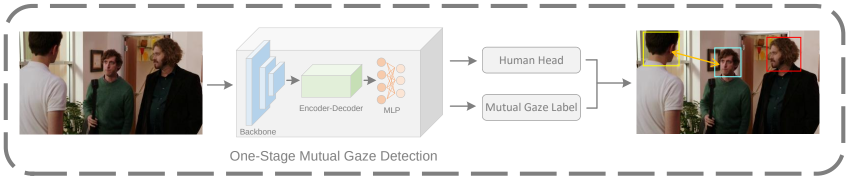


**Fig. 2.** An illustration of proposed one-stage method. Utilizing the designed mutual gaze instance triples, our proposed one-stage method can detect all human heads and the corresponding mutual gaze labels in parallel, which accelerates the pipeline of mutual gaze detection.

on the global image information. By designing mutual gaze instance triples, we improve the mutual gaze detection process from serial to parallel, which greatly accelerates the inference speed without losing performance.

Our proposed MGTR consists of four following modules: A Backbone for feature extraction, Transformer Encoder, Transformer Decoder, and a Fully Connected Neural Network for mutual gaze instance prediction. First, a convolutional neural network is used to extract image features, then a one-dimensional vector is generated by flattening the feature map and we combine it with the positional encoding [22,4] to get the input of Encoder. After that, the output of Encoder combined with learnable mutual gaze queries are passed through Decoder to model connections between different people, and finally, the encoded mutual gaze queries are passed through a fully connected neural network to output the mutual gaze instance as the result of our model.

Overall, our main contributions are as follows:

- We build a *one-stage* model called MGTR which combines the human head detection and the mutual gaze recognition. To the best of our knowledge, this is the first work that integrates mutual gaze detection task into a one-stage method.
- Modeling the location information of people and the relationship between them using global information instead of head image crops.
- Our model outperforms the state-of-the-art method on end-to-end mutual gaze detection task. Moreover, MGTR can perform faster mutual gaze detection.

## 2 Related Work

In this section, we first review methods for gaze estimation which encompass a variety of gaze types (Section 2.1). Then go down to the literature of mutual gaze detection (Section 2.2). At last one-stage detection methods (Section 2.3) will be reviewed.

### 2.1 Gaze Estimation in Social Scenarios

Eye gaze can convey rich information and is closely related to the attention, intention, and emotion of a person, even people from different cultures may share a similar meaning of eye gaze [10]. Recently, in the computer vision community, there are also a lot of research focusing on social scenario gaze estimation and yielding promising results. For example, Lian *et al* [13] proposed a solution for gaze point prediction of the target persons. Fan *et al* [7] proposed a spatial-temporal modeling method to detect people's looking at the same target simultaneously. Zhuang *et al* proposed MUGGLE [28], an approach that is suitable for massive people's shared-gazing. In order to detect whether two people in the video are looking at each other, Marin *et al* [20,19] proposed a method based on the spatial and temporal information to solve this problem. To understand different types of eye gaze in a group of people, Fan *et al* [8] proposed a method to detect multiple types of human gaze, such as single gaze, shared gaze, etc. In this work, we focus on image based one-stage mutual gaze detection.

### 2.2 Mutual Gaze Detection

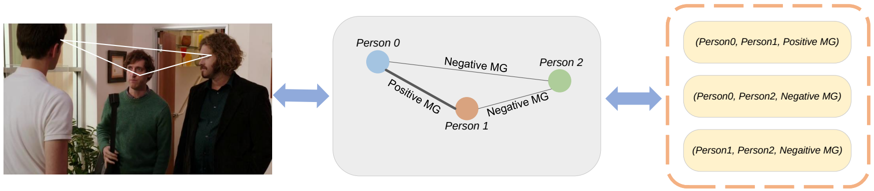
Mutual gaze is one of the most common types of social scenarios gaze communication and there are also methods trying to achieve automatic mutual gaze detection. These methods are all two-stage methods consisting of a human head detector in the first stage and a mutual gaze classifier in the second stage. Specifically, Marin *et al* proposed viode-based LAEO-Net [20] and get a promising result in mutual gaze detection by considering both temporal and spatial information. After that, they further modified LAEO-Net to get LAEO-Net++ [19], which achieved better performance. Moreover, the image-based approach proposed by Doosti *et al* [6] takes advantage of multi-task learning by using pseudo 3D gaze to boost mutual gaze detection. However, these methods suffer from the lack of global image information and slow inference speed due to the sequential two-stage architecture.

### 2.3 One-stage Detection Method

Recently, in the field of computer vision, many research designs have followed a one-stage idea to speed up the processing pipeline. For example, in the field of object detection, the favorable results of SSD [16], YOLO [23], RetinaNet [14], DETR [5] and other methods have demonstrated the advantages of one-stage detection methods. Compared with two-stage detection methods, one-stage methods can perform the detection and classification tasks by using only one network and the pipeline is generally simpler, faster, and more computationally efficient so that it is easier to adopt for real-world applications.

## 3 Method

The task of one-stage mutual gaze detection is to give an image and then detect all mutual maze instances in the image in an end-to-end way. In this section,


Person 0
Negative MG
Person 2
Positive MG
Negative MG
Person 1
(Person0, Person1, Positive MG)
(Person0, Person2, Negative MG)
(Person1, Person2, Negaitive MG)

(a) Mutual gaze instance in real world

(b) Graph abstracted from the corresponding real scene

(c) Mutual gaze representation to the scene

**Fig. 3.** Representation of Mutual Gaze Instances in the scene. In the social scene, each individual is uniquely numbered and the enumerated head pairs are order-independent. *MG* means Mutual Gaze.

we will first describe the Representation of Mutual Gaze Instances (Section 3.1) followed by the detailed Model Architecture (Section 3.2), after that we will introduce the Strategy for Mutual Gaze Instances Match (Section 3.3) and at last the Loss Function Setting (Section 3.4).

### 3.1 Representation of Mutual Gaze Instances

We define a mutual gaze instance as a triple, namely (Person1, Person2, Mutual Gaze Label), where Person1 and Person2 contain the bounding box coordinates and the class confidence of a person, and Mutual Gaze Label is one when Person1 and Person2 are looking at each other otherwise zero. It is noteworthy that under the mutual gaze detection task, it seems useless to predict the class of each box, however, this setting can be used as a detection threshold when we conduct the test phase in which we need the person class confidence of each box. A detailed example of representing a real scene with a mutual gaze instance is given in Fig.3. Additionally, a mutual gaze instance is unordered, that is to say, the relationship between $i$-th Person and $j$-th Person only needs to be recorded once as (Person$i$, Person$j$, Mutual Gaze label between $i$ and $j$).

### 3.2 Model Architecture

Our proposed MGTR mainly consists of four parts: a Backbone, an Encoder module, a Decoder module, and MLP. Fig.4 shows an overview of MGTR architecture.

**Backbone** A convolutional neural network is used to extract features from an input image of original size $[H, W, 3]$. After the convolutional neural network, we get a feature map of size $[C, H, W]$. We then reduce the channel dimension of the feature map from $C$ to $d$ by a $1 \times 1$ convolution kernel resulting the new feature map of size $[d, H, W]$. Since Transformer Encoder requires a sequence as input data, we compress the last two dimensions of the new feature map to obtain a flatten feature called *input embedding* of size$[d, HW]$.

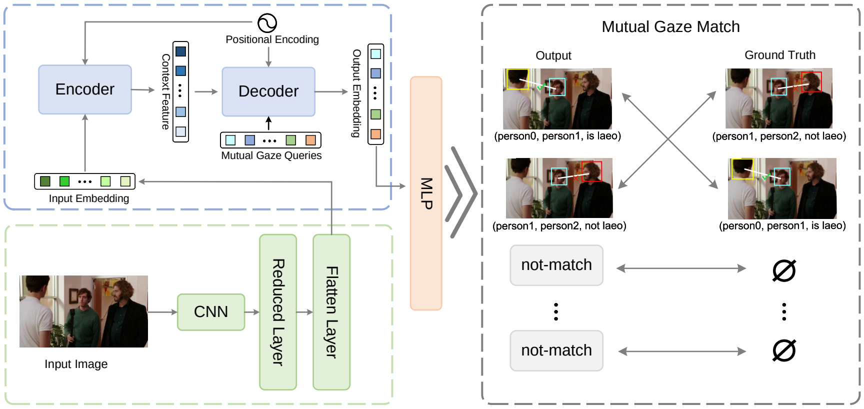


**Fig. 4.** An overview of MGTR architecture. It consists of four components: a Backbone to convert input image into an one-dimensional input embedding, an Encoder followed by a Decoder to get the encoded Mutual Gaze Queries, and MLP to predict mutual gaze instances. *laeo* means Looking At Each Other. See Section 3.2 for more details.

**Encoder** The Encoder layer in MGTR is the same as the standard Transformer Encoder layer, including a multi-head self-attention layer and a feed forward network(FFN). Due to the permutation invariance of the Transformer, we add the *positional encoding* to the input embedding to obtain the Query and Key of Encoder and only use the input embedding as the Value. For the convenience of description, we represent the output of the Encoder as the *context feature*.

**Decoder** The Decoder layer in MGTR is also the same as the standard Transformer Decoder layer which contains two multi-head attention layers and a feed-forward network. We refer to the $N$ learnable positional embeddings as *mutual gaze queries*. In the multi-head self-attention layer, the Query, Key, and Value all come from either the mutual gaze queries or the sum of the previous decoder layer's output and mutual gaze queries. As for the encoder-decoder cross attention layer, the Value comes from the context feature generated from the Encoder, the Key is the sum of context feature and positional encoding and the Query is the sum of the multi-head self-attention layer's output and mutual gaze queries. We denote the output of the Decoder as *output embedding*.

The self-attention mechanism in the Encoder and Decoder can help us model the positions of different people in the image and the relationship between them. The $N$ output embeddings encoded from $N$ mutual gaze queries are then converted into a mutual gaze instance by the subsequent MLP so that we get $N$ final mutual gaze instance results and we will discuss this part next.

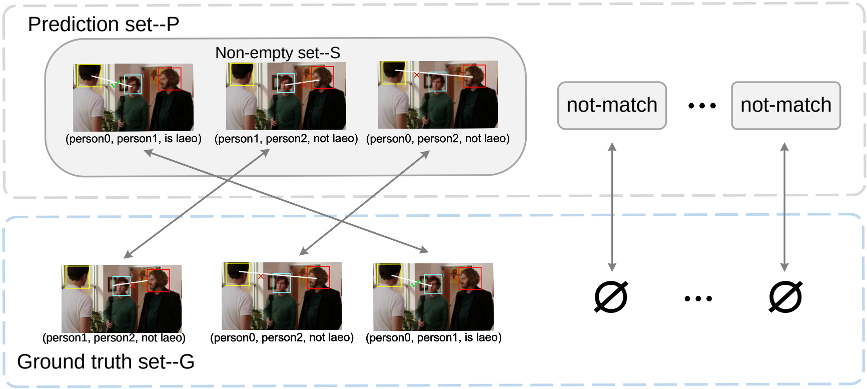

Prediction set--P
Non-empty set--S
(person0, person1, is laeo)
(person1, person2, not laeo)
(person0, person2, not laeo)
not-match
...
not-match
(person1, person2, not laeo)
(person0, person2, not laeo)
(person0, person1, is laeo)
∅
...
∅
Ground truth set--G

**Fig. 5.** An example explaining the mutual gaze instances match. By padding the set **G** with ∅, we make **P** and **G** the same size, which can be transformed into a bipartite graph matching problem. *laeo* means Looking At Each Other

**MLP for Mutual Gaze Prediction** After passing the $N$ mutual gaze queries through the Decoder, we get $N$ output embedding, which contains information about the position of the head bounding boxes and the relationship between different people in the image. Then we pass the output embedding into the MLP to predict the mutual gaze instance. Specifically, three one-layer fully-connected neural networks are used to predict the confidence score of Person1, Person2, and Mutual Gaze Label respectively, and two three-layer fully-connected neural networks are used to predict the head bounding boxes of Person1 and Person2. For the human confidence score prediction branch, there are three classes: whether it is a person and *not-match* (the meaning of not-match will be described later). For the mutual gaze confidence prediction branch, there are also three categories that indicate whether Person1 and Person2 are looking at each other and not-match. We then apply Softmax to the results of all confidence prediction branches to obtain normalized confidence. For the branch of head bounding boxes regression, the output dimension of the MLP is four, which represent the normalized center coordinates, width, and height of the bounding boxes respectively.

### 3.3 Strategy for Mutual Gaze Instances Match

After passing the output embedding through MLP, we get $N$ predicted mutual gaze instances. However, the number of ground truth mutual gaze instances is not necessarily $N$ (often less than $N$). So it requires some predicted mutual gaze instances representing empty which we denoted as *no-match*, indicating that these mutual gaze instances do not match any ground truth. To be precise, we denote the set of ground truth instances as **G**, and the size of **G** is $M$, the set of predicted instances is **P** and the size of **P** is $N$, then a satisfactory model should output a set which contains $N - M$ instances representing *not-match*.

After solving the problem that the number of ground truth instances and predicted instances are not always equal by designing the *not-match* class, the next key problem is how to match the predicted instances with the ground truth instances. Specifically, we denote the set of elements predicted to be non-empty in **P** as **S**, then the problem we have to solve is how to build a map $\sigma$ from **S** to **G**. It is worth mentioning that the size of **S** is not necessarily equal to the size of **G**. We solve this matching problem in another way: by padding the ground truth set **G** with $\varnothing$, we make **G** and **P** equal in size. So the above matching problem is transformed into a one-to-one bipartite matching problem between **P** and **G**. In this work, we use the Hungarian algorithm [12] to solve this problem. A more concrete example can be seen in Fig.5.

### 3.4 Loss Function Setting

Assume the mapping from the predicted set **P** to the ground-truth set **G** is denoted as $\sigma(i)$, which means the $i$-th element in the set **P** will be mapped to the $\sigma(i)$-th element in **G**. We design the matching cost of the Hungarian algorithm as follows.

$$\mathcal{L}_{match} = \sum_{i=1}^{N} [\beta_1 \mathcal{L}_{class}(c_i, c_{\sigma(i)}) + \beta_2 \mathcal{L}_{box}(b_i, b_{\sigma(i)})] \tag{1}$$

where $\mathcal{L}_{class}(c_i, c_{\sigma(i)})$ represents the cost between the $i$-th mutual gaze instance from **P** and the $\sigma(i)$-th ground truth from **G** in human class confidence and mutual gaze confidence, and we call it the *class loss function.* $\mathcal{L}_{box}(b_i, b_{\sigma(i)})$ represents the cost between the $i$-th mutual gaze instance from **P** and the $\sigma(i)$-th from **G** in head bounding boxes regression, and we call it the *head bounding box regression loss function.* $\beta_1$ and $\beta_2$ are hyperparameters used to measure the weight between these two types of losses. The specific forms of these two losses are discussed below.

For the class loss function, we denote the value of mutual gaze confidence as $p_i^{gaze}$ and use $p_i^{h_1}$ and $p_i^{h_2}$ to represent the human class confidence. Under this representation, the class loss function is defined as follow.

$$\mathcal{L}_{class}(c_i, c_{\sigma(i)}) = \alpha_1 p_i^{h_1} + \alpha_2 p_i^{h_2} + \alpha_3 p_i^{gaze} \tag{2}$$

For the head bounding box regression loss function, the definition is as follow.

$$\mathcal{L}_{box}(b_i, b_{\sigma(i)}) = \gamma_1 \ell_1(b_i, b_{\sigma(i)}) + \gamma_2 \mathrm{GIoU}(b_i, b_{\sigma(i)}) \tag{3}$$

where $\ell_1(\cdot)$ is the $L_1$ loss. We also added the GIoU loss [24] into the head bounding box regression loss function, and we will confirm the importance of GIoU loss in the later Ablation Study.

By defining the above costs from the $i$-th element in **P** to the $\sigma(i)$-th element in **G**, we can solve the following optimal bipartite match problem based on the Hungarian algorithm.

$$\sigma^* = \operatorname*{argmin}_{\sigma} \mathcal{L}_{match} \tag{4}$$

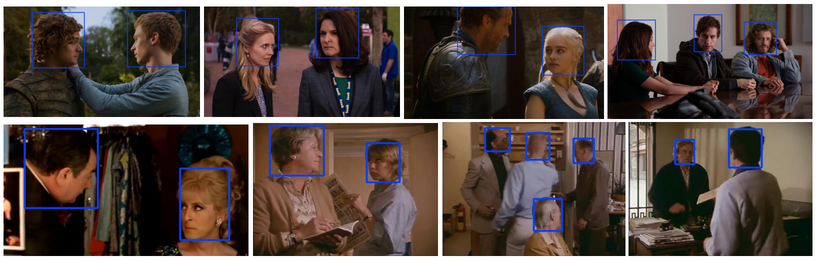


**Fig. 6.** Some examples of UCO-LAEO and AVA-LAEO datasets. The images in the first row come from the UCO-LAEO dataset which contains both head bounding boxes and mutual gaze labels. The images in the second row are from the AVA-LAEO dataset which only contains mutual gaze labels and we approximate the ground-truth head bounding box coordinates by using a pre-trained head detector [16].

After the optimal bipartite matching problem is solved by the Hungarian algorithm, we can next calculate the loss function for training. In this work, the definition of loss function is almost the same as in Eq.1, the difference is that in $\mathcal{L}_{class}$ we use cross-entropy loss instead of negative predicted probability and there are also some subtle differences in the settings of hyperparameters as well (more details can be seen in Section 4.4).

Different from previous mutual gaze detection methods which first train a head detector, and then freeze the head detector parameters and train the mutual gaze classifier. The method in our work optimizes the class loss function and the head bounding box regression loss function simultaneously during the model training process.

## 4 Experiments

### 4.1 Datasets

**UCO-LAEO [20]** This dataset consists of detailed mutual gaze instance annotations with human head bounding boxes and mutual gaze labels. The original dataset has been manually divided into positive and negative instances for sample balance, which means there may be some instances in one image that are not annotated as negative instances. In this work, since we need to detect all the instances in one image, so we only use positive samples from original annotations and treat all remaining instances as negative samples.

**AVA-LAEO [20]** This dataset has a broad coverage that reaching 50,797 video frames generated from *Atomic Visual Actions* dataset(AVA v2.2) [21]. Since there are no manual head bounding box annotations in the original AVA v2.2 dataset, we first use a pre-trained head detector [16] to generate all head bounding box coordinates in each frame as head bounding box ground truth before our training starts. Similarly, we also only use positive instances from the original dataset annotations and regard all the remaining as negative.

Some examples of UCO-LAEO and AVA-LAEO datasets are shown in Fig.6.

### 4.2 Evaluation Metric

In this work, we use the *mean average precision*(mAP) as a measure of how well our model performs on the two datasets. It is worth mentioning that although our task is similar to a binary classification task, we do not use the AP of a single class as an evaluation metric. This is because our task is to simultaneously predict accurate human head positions as well as mutual gaze labels. So we need to consider both classes of AP to include the evaluation of head position localization accuracy. For example, if it is known that the model predicts the mutual gaze label in the current instance to be positive but in fact it is a wrong prediction, it does not necessarily mean that the ground-truth mutual gaze label is negative because it is also possible that the model does not match the head bounding box with ground truth when predicting. So, an instance predicts correctly if and only if it locates the head boxes of the two people in the correct position and predicts correctly whether the two are looking at each other. This criterion requires both the AP in two classes should be high.

### 4.3 Current State-of-the-Art Approach

The current state-of-the-art approach for video-based mutual gaze detection task is LAEO-Net++ proposed by Marin *et al* [19]. The image-based state-of-the-art method is proposed by Doosti *et al* [6] and we call it *Pseudo 3D Gaze* in this paper.

The two aforementioned works are different from ours. Specifically, both works above only detect mutual gaze without detecting human head position, while MGTR detects above both whose task is more difficult. Moreover, LAEO-Net++ is a video-based method which uses ground-truth human head box as model input and utilizes temporal connection among neighbor video frames resulting in the input containing more prior information, while ours is an image-based method whose input is only one single image. Since these differences, we modify them accordingly for a fair comparison. Specifically, as for LAEO-Net++, since we focus on image-based mutual gaze detection in this work, so we plan to use its image-based version introduced in [6]. However, we cannot get the modified LAEO-Net++ for comparison since the code in [6] is not open source. So we directly use the performance numbers of image-based *LAEO-Net* reported in [6]. As for Pseudo 3D Gaze which focuses on the performance of the second stage, we add a head detector in front of the original model for end-to-end mutual gaze detection, a detailed description of this model can be seen in Section 4.4.

### 4.4 Implementation Details

**Data Augmentation** We normalize the input image by using the mean and std from ImageNet [11]. To improve the robustness of the model, we randomly apply horizontal flipping, adjusting brightness and contrast, random cropping, and random resize (to enable the model to detect instances at multiple scales).

**Hyperparameters Settings** In order to balance the class cost and head bounding box regression cost, we set $\beta_1 = 1.2$, $\beta_2 = 1.0$ in the Hungarian algorithm matching process and $\beta_1 = \beta_2 = 1.0$ in the training loss function. At the same time, we set $\alpha_1 = \alpha_2 = 1.0$, $\alpha_3 = 2.0$ both in the Hungarian algorithm and loss function to make our model focus more on judging the existence of mutual gaze. In the head bounding box regression loss function, we set $\gamma_1 = 5.0$, $\gamma_2 = 2.0$.

**Training Settings** We used Resnet50 [9] with frozen batchnorm layer as Backbone for MGTR, the number of Encoder and Decoder layers are both set to 6, the same as in [29]. Both Backbone and Encoder-Decoder use the pretrained parameters from COCO [15] pretrained DETR [5]. The number of mutual gaze queries is set to 100. In the training phase, we choose the batch size to be 8, and we use AdamW [18] as an optimizer, with a constant learning rate of 1e-4 in Encoder-Decoder and 1e-5 in Backbone, we train the model until the performance on the test set no longer improves.

**Description for Pseudo 3D Gaze using Head Detector** Since the Pseudo 3D Gaze method in [6] uses the ground truth head bounding boxes, for a fair comparison, we set this baseline that consists of two parts: a head detector and a mutual gaze classifier. We use the head detector proposed in [26] for head box detection. The classifier of this baseline adopts the same network as the one proposed by Doosti *et al* and also uses the pseudo 3D gaze to boost the training process. During training, we only train the mutual gaze classifier with randomly initialized weights by using the ground truth head bounding boxes. During testing, we first detect each head bounding boxes through the pretrained detector and then pass the paired detected head crops through the mutual gaze classifier to get the result for each mutual gaze instance.

### 4.5 Comparison with State-of-the-Art Method

Table 1 shows the quantitative results compared with the state-of-the-art method in terms of FPS and mAP on UCO-LAEO and AVA-LAEO datasets.

For UCO-LAEO dataset, MGTR processing is more than 150 times faster than the two-stage approach in extreme social scenes with more than four people and nearly 17 times faster in scenes averaged across the test set. Moreover, MGTR achieves 64.8% mAP, an 18.1% increase over the Pseudo 3D Gaze using detected head bounding boxes. Meanwhile, our method is also comparable with image-based SoTA methods who use ground truth head bounding boxes, with only a difference of 0.3%. The good performance of MGTR on mAP shows that MGTR can handle the imbalance of positive and negative samples in the training set well.

**Table 1.** Comparison with State-of-the-Art Method. $\text{FPS}_{\text{ext}}$ refers to the number of images processed per second in the extreme social scene with more than four people, and $\text{FPS}_{\text{all}}$ refers to the number of pictures processed per second in scenes averaged across the whole test set. The FPS is evaluated using NVIDIA 3090TI GPU. Since we have no access to the code of image based LAEO-Net in [6], we do not evaluate the FPS of image based LAEO-Net. w/ GT and w/t GT respectively indicate whether to use ground truth head bounding boxes as model input. The mAP represents positive class's AP in two-stage methods and two classes's average AP in one-stage methods. *Number reported from [6].

|  |  |  | $\text{FPS}_{\text{ext}}$ |  | $\text{FPS}_{\text{all}}$ |  | mAP |  |  |  |
|---|---|---|---|---|---|---|---|---|---|---|
|  | Method | End-to-end training |  |  |  |  | UCO-LAEO |  | AVA-LAEO |  |
|  |  |  | UCO | AVA | UCO | AVA | w/ GT | w/t GT | w/ GT | w/t GT |
| $Two-stage^*$ | Iamge based LAEO-Net | ✗ | - | - | - | - | 55.9 | - | 70.2 | - |
|  | Pseudo 3D Gaze | ✗ | 0.51 | 0.93 | 0.49 | 0.89 | **65.1** | 46.7 | **72.2** | 52.3 |
| $One-stage$ | MGTR (ours) | ✓ | **78.06** | **14.56** | **9.18** | **10.81** | - | **64.8** | - | **66.2** |

As for AVA-LAEO dataset, MGTR processes each image more than 15 times faster than the two-stage method in more than four people social scenes and more than 12 times in the average scene across the test set. Besides, MGTR gets a 66.2% mAP score, a 13.9% increase compared with the baseline using detected head bounding boxes.

### 4.6 Ablation Study

In this part, we design some ablation methods to study how data augmentation, different components of MGTR and loss function setting will affect the performance. We choose the UCO-LAEO dataset and use MGTR with Resnet50 backbone as a base model. Ablation baselines are as follows: (1) No multi-scale and random cropping in data augmentation (2) Using Resnet101 as a backbone, we design this part to study how the complexity of the model will affect performance (3) No GIoU loss, we only use BCE Loss and $L_1$ Loss as loss function. (4) Use DIoU loss [27] instead of original GIoU loss and keep other parts consistent with base model. We design this part to explore the effect of different IoU losses. (5) Use CIoU loss [27] to replace GIoU loss and keep other parts unchanged.

The results of the ablation study are provided in Table 2. It can be seen that data augmentation including multi-scale resize and random cropping are important for training, without which resulting in an 11.4% drop in mAP. At the same time, the use of Resnet101 as the backbone leads to a decrease on performance, so the more complex the model is not always the better. As can be seen from the fourth-to-last row of Table 2, when the GIoU loss is removed from the loss function, the performance of MGTR on mAP drops by 9.0%, which indicates that the GIoU loss is indispensable for MGTR to accurately locate each person's head bounding box. When using DIoU loss or CIoU loss to replace the original GIoU loss, the performance on mAP drops by 9.1% and 4.4%, respectively, indicating that even though DIoU and CIoU are improvements over

**Table 2.** Ablation Study of MGTR. mAP refers to the average AP in the two classes, $\mathrm{AP}_{\mathrm{rare}}$ refers to the AP of the minority category (usually the positive mutual gaze label), and $\mathrm{AP}_{\mathrm{normal}}$ refers to the majority category, and Recall refers to the average Recall over two classes.

| Model Setting | #param | mAP | $\mathrm{AP}_{\mathrm{rare}}$ | $\mathrm{AP}_{\mathrm{normal}}$ | Recall |
|---|---|---|---|---|---|
| NoDataAugmentation | 41.4M | 53.4 | 52.8 | 54.3 | 68.6 |
| Resnet101Backbone | 60.3M | 51.3 | 49.7 | 53.0 | 65.5 |
| NoGIoULoss | 41.4M | 55.8 | 52.8 | 58.8 | 67.3 |
| DataAug+Resnet50+DIoU | 41.4M | 55.7 | 51.1 | 60.4 | 69.2 |
| DataAug+Resnet50+CIoU | 41.4M | 60.4 | 54.5 | 66.3 | 68.4 |
| Base(DataAug+Resnet50+GIoU) | 41.4M | **64.8** | **58.3** | **71.4** | **75.3** |

GIoU, using GIoU loss still achieves the best performance. This may be due to the fact that both Backbone and Encoder-Decoder in MGTR are initialized using the parameters in DETR pretrained model that also uses GIoU loss. Therefore, using a consistent loss may give better results.

### 4.7 Qualitative Analysis

Different from the previous two-stage methods, MGTR simultaneously gives all head bounding boxes and mutual gaze labels in the scene in an end-to-end manner, which does not seem easy to understand. In this part, we will analyze the different roles of Encoder and Decoder of MGTR in different levels of image semantic understanding.

To study the different roles of Encoder and Decoder in MGTR, we visualize the last attention layer of Encoder and Decoder respectively, results can be seen in Fig.7. It can be easily seen that the role of Encoder in MGTR is to find all the head bounding boxes in the social scene, because the head area of each person in the attention-map is given a larger attention weight. After the Encoder finds all the people in the scene, Decoder can find the relationship between different people. In the Decoder's attention-map, we can see that when the mutual gaze label is positive there will be two people's head regions with large attention weights. However, when the label is negative, only one person's head region will be focused. Therefore, we can conclude that the role of Decoder is to predict which pairs of people in the current scene are looking at each other so that model the relationship between different people.

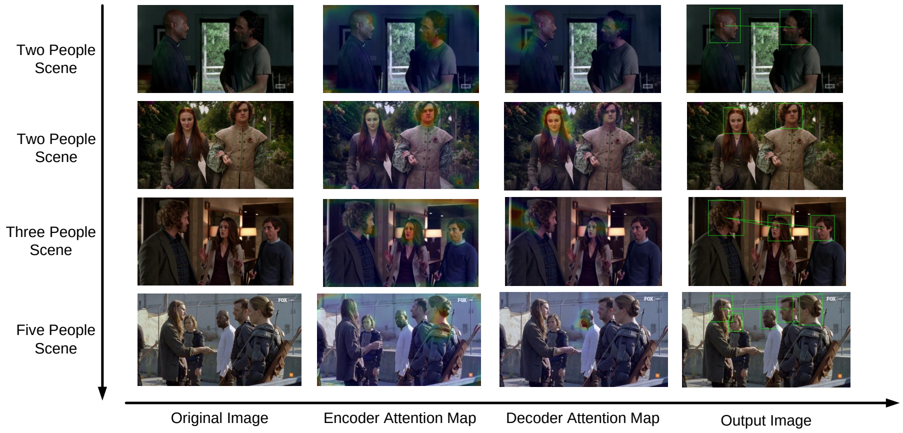


**Fig. 7.** Visualization of last attention layer in Encoder and Decoder and the predicted result by MGTR (from UCO-LAEO dataset). For each row, the first image is the original input image, the second image is the attention weight in Encoder, the third image is the attention weight in Decoder, and the last image is the predicted result by MGTR.

## 5 Conclusion

In this work, we propose a one-stage mutual gaze detection method called Mutual Gaze TRansformer or MGTR to directly predict mutual gaze instances in an end-to-end manner. Different from current two-stage mutual gaze detection methods, MGTR is the first work that integrates human head detection and mutual gaze recognition into one stage which simplifies the detection pipeline. Experiments on two mutual gaze datasets demonstrate that our proposed method can greatly accelerate inference process while improving performance. In the future, we will explore the application of incorporating mutual gaze information into the analysis of the interpersonal relationship, and the detected mutual gaze instance will serve as an important clue for social scene interpretation.

## References

1. Abele, A.: Functions of gaze in social interaction: Communication and monitoring. Journal of Nonverbal Behavior **10**(2), 83–101 (1986)
2. Admoni, H., Scassellati, B.: Social eye gaze in human-robot interaction: a review. Journal of Human-Robot Interaction **6**(1), 25–63 (2017)
3. Argyle, M., Cook, M.: Gaze and mutual gaze. (1976)
4. Bello, I., Zoph, B., Vaswani, A., Shlens, J., Le, Q.V.: Attention augmented convolutional networks. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 3286–3295 (2019)

5. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: European conference on computer vision. pp. 213–229. Springer (2020)
6. Doosti, B., Chen, C.H., Vemulapalli, R., Jia, X., Zhu, Y., Green, B.: Boosting image-based mutual gaze detection using pseudo 3d gaze. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 35, pp. 1273–1281 (2021)
7. Fan, L., Chen, Y., Wei, P., Wang, W., Zhu, S.C.: Inferring shared attention in social scene videos. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6460–6468. IEEE (2018)
8. Fan, L., Wang, W., Huang, S., Tang, X., Zhu, S.C.: Understanding human gaze communication by spatio-temporal graph reasoning. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5724–5733 (2019)
9. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
10. Kleinke, C.L.: Gaze and eye contact: a research review. Psychological bulletin **100**(1), 78 (1986)
11. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. Communications of the ACM **60**(6), 84–90 (2017)
12. Kuhn, H.W.: The hungarian method for the assignment problem. Naval research logistics quarterly **2**(1-2), 83–97 (1955)
13. Lian, D., Yu, Z., Gao, S.: Believe it or not, we know what you are looking at! In: Asian Conference on Computer Vision. pp. 35–50. Springer (2018)
14. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. In: Proceedings of the IEEE international conference on computer vision. pp. 2980–2988 (2017)
15. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: European conference on computer vision. pp. 740–755. Springer (2014)
16. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., Berg, A.C.: Ssd: Single shot multibox detector. In: European conference on computer vision. pp. 21–37. Springer (2016)
17. Loeb, B.K.: Mutual eye contact and social interaction and their relationship to affiliation (1972)
18. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. arXiv preprint arXiv:1711.05101 (2017)
19. Marín-Jiménez, M.J., Kalogeiton, V., Medina-Suárez, P., , Zisserman, A.: LAEO-Net++: revisiting people Looking At Each Other in videos. IEEE Transactions on Pattern Analysis and Machine Intelligence (2021). https://doi.org/10.1109/TPAMI.2020.3048482
20. Marin-Jimenez, M.J., Kalogeiton, V., Medina-Suarez, P., Zisserman, A.: Laeo-net: revisiting people looking at each other in videos. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3477–3485 (2019)
21. Murray, N., Marchesotti, L., Perronnin, F.: Ava: A large-scale database for aesthetic visual analysis. In: 2012 IEEE conference on computer vision and pattern recognition. pp. 2408–2415. IEEE (2012)
22. Parmar, N., Vaswani, A., Uszkoreit, J., Kaiser, L., Shazeer, N., Ku, A., Tran, D.: Image transformer. In: International Conference on Machine Learning. pp. 4055–4064. PMLR (2018)

23. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, real-time object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 779–788 (2016)
24. Rezatofighi, H., Tsoi, N., Gwak, J., Sadeghian, A., Reid, I., Savarese, S.: Generalized intersection over union: A metric and a loss for bounding box regression. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 658–666 (2019)
25. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017)
26. Zhang, S., Zhu, X., Lei, Z., Shi, H., Wang, X., Li, S.Z.: S3fd: Single shot scale-invariant face detector. In: Proceedings of the IEEE international conference on computer vision. pp. 192–201 (2017)
27. Zheng, Z., Wang, P., Liu, W., Li, J., Ye, R., Ren, D.: Distance-iou loss: Faster and better learning for bounding box regression. In: Proceedings of the AAAI conference on artificial intelligence. vol. 34, pp. 12993–13000 (2020)
28. Zhuang, N., Ni, B., Xu, Y., Yang, X., Zhang, W., Li, Z., Gao, W.: Muggle: Multi-stream group gaze learning and estimation. IEEE Transactions on Circuits and Systems for Video Technology **30**(10), 3637–3650 (2019)
29. Zou, C., Wang, B., Hu, Y., Liu, J., Wu, Q., Zhao, Y., Li, B., Zhang, C., Zhang, C., Wei, Y., et al.: End-to-end human object interaction detection with hoi transformer. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11825–11834 (2021)