OmniGF: A Dual-Branch Vision-Language Framework for Unified Gaze Following

Qiaomu Miao[1] **Haoyu Wu**[1] **Jingyi Xu**[1] **Minh Hoai**[2] **Dimitris Samaras**[1]
[1]Stony Brook University [2] Adelaide University

Abstract

Understanding human gaze behavior is essential for complex scene comprehension and human-computer interaction. Traditional gaze following models are typically restricted to pure spatial localization, lacking the high-level capacity to reason about semantic targets or complex social contexts. Furthermore, these models often process individuals sequentially, requiring redundant computations over the same scene image for multi-person inference. While recent Vision-Language Models (VLMs) offer the exceptional semantic reasoning needed to address gaze-related semantic tasks, their reliance on discrete text generation inherently limits precision in continuous spatial tasks like gaze localization. To bridge this gap, we propose OmniGF, a unified vision-language framework that adapts foundational VLMs for scalable multi-person gaze reasoning. The model adopts a dual-branch decoding strategy: a structured language branch generates discrete reasoning states, while a continuous spatial branch directly taps into the VLM's dense hidden states. Supervising these extracted representations with high-resolution gaze target heatmaps effectively overcomes the spatial bottleneck of text-only coordinate generation. Furthermore, to explicitly ground the model in multi-person scenes, we augment the input with head embeddings encoded from cropped head images, providing fine-grained appearance and orientation cues for all individuals simultaneously. By modeling all individuals and leveraging the strong semantic capability of VLMs, OmniGF seamlessly integrates precise spatial gaze target estimation, semantic gaze prediction, and complex social gaze reasoning. Extensive experiments demonstrate that our framework establishes new state-of-the-art performance across multiple standard benchmarks. Code is available at `https://github.com/cvlab-stonybrook/omnigf`.

1 Introduction

Gaze following, which aims to infer where a person is looking within a scene, is a fundamental component of human social cognition and a critical task in computer vision. Accurately estimating human gaze targets is essential for a wide array of downstream applications, including human-robot interaction [1–3], autonomous driving [4–6], social behavioral analysis [7, 8], and the diagnosis of neurodevelopmental disorders [9, 10]. Historically, the field has been dominated by specialized Convolutional Neural Networks (CNNs) [11–14] and Vision Transformers (ViTs) [15, 16]. While effective for basic spatial localization, these traditional models often process individuals sequentially, requiring redundant processing of the same scene image to predict gaze targets for multiple individuals. Furthermore, most methods are limited to pure spatial localization, lacking the high-level capacity to reason about the target semantics or to understand the broader context of human social interactions.

Recently, large Vision-Language Models (VLMs) have demonstrated unprecedented capabilities in generalized visual reasoning, achieving remarkable success across a variety of zero-shot and downstream tasks related to semantic reasoning, such as referring expression comprehension [17] and semantic segmentation [18, 19]. However, trivially applying off-the-shelf VLMs to gaze following yields clearly suboptimal performance. Standard VLMs represent spatial coordinates using discrete

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

language tokens, a paradigm that is inherently misaligned with the continuous, high-resolution precision required for dense spatial tasks like gaze localization. A concurrent work, Eyes on VLM [20], shows that VLMs predicting gaze targets as text coordinates still lag behind specialized vision-based gaze models, even after task-specific fine-tuning. While a few pioneering works have attempted to bridge this gap, they remain limited. For instance, GazeVLM [21] relies on auxiliary depth map inputs yet fails to achieve competitive spatial precision against state-of-the-art vision-only models. Similarly, VL4Gaze [22] predominantly focuses on Gaze-oriented Visual Question Answering (VQA) and does not comprehensively address or evaluate standard 2D spatial gaze following benchmarks. Consequently, effectively harnessing the semantic power of VLMs for precise, multi-person gaze localization remains an open challenge.

To overcome these limitations, we propose OmniGF, a novel, unified Vision-Language framework for multi-person gaze target estimation and reasoning. Instead of forcing a VLM to regress spatial coordinates purely through discrete text tokens, we introduce a dual-branch decoding architecture. Our framework utilizes a fine-tuned structured language branch to generate discrete reasoning states (e.g., 2D target location, target in/out of frame, and semantic category), while simultaneously employing a continuous gaze branch dedicated to the dense spatial decoding of target heatmaps. This spatial decoder dynamically taps into the VLM's hidden state tokens derived from the generated gaze-related text for each person. By associating the visual embeddings of the scene with the gaze representations from the hidden states for each person, the model predicts gaze target heatmaps that directly benefit from the dense supervision signals, significantly improving spatial localization performance.

Furthermore, to ensure more efficient processing of complex scenes, OmniGF models multiple individuals jointly. To explicitly ground the VLM to each specific individual, we introduce a head-conditioned token injection mechanism. By embedding cropped visual head features directly into dedicated prompt token slots, we provide the backbone with the fine-grained appearance and orientation cues necessary to accurately disambiguate targets in dense environments. Crucially, by leveraging the strong semantic reasoning capacity of the VLM backbone, our framework seamlessly extends beyond simple coordinate regression. OmniGF jointly supports auxiliary tasks such as semantic gaze target recognition (identifying *what* object is being looked at) and pairwise social gaze reasoning (e.g., shared attention and mutual gaze), unifying spatial, semantic, and social gaze understanding within a single model.

In summary, our main contributions are as follows:

• **Unified Multi-Person VLM Framework.** We introduce OmniGF, a person-aware vision-language framework that models multiple individuals jointly via structured prompts, processing all persons together from a single VLM input.

• **Head-Conditioned Token Injection.** We propose a simple yet effective mechanism that injects head-centric visual embeddings into dedicated token slots, grounding the VLM with fine-grained appearance and orientation cues for each individual.

• **Dual-Branch Decoding for Reasoning and Spatial Prediction.** We design a dual-branch architecture that combines structured language generation for discrete reasoning with a gaze branch that extracts hidden states to predict dense, high-resolution gaze heatmaps.

• **Unified Gaze, Semantic, and Social Understanding.** Our framework jointly supports gaze target estimation, semantic gaze prediction, and pairwise social gaze reasoning within a single model, achieving state-of-the-art performance across five different benchmarks.

2 Related Works

Gaze Following aims to predict the location where a person is looking in a scene image. Early methods mostly adopted a two-branch design for gaze target prediction, with one branch encoding head pose and appearance and the other encoding scene saliency information [11, 23, 12, 24]. Later works leveraged additional modalities, such as depth and human pose, to further improve model performance [13, 25, 14, 26–30]. More recent models have incorporated large pretrained foundation models and achieved impressive performance on gaze following. Sharingan [15] uses Multi-MAE [31] for multi-person gaze following, while Gaze-LLE [16] employs DINOv2 [32] with a head-prompting strategy and achieves further improvements. In parallel, several works have addressed gaze following from different perspectives, such as jointly predicting subject heads and their corresponding gaze

targets [33–35], incorporating audio cues for gaze following [36], or estimating gaze targets from multiple camera views [37]. Meanwhile, recent works have explored jointly solving additional tasks together with gaze target estimation. Tafasca et al. [38] explored joint prediction of the location and semantic label of the gaze target, while Gupta et al. [39] studied joint gaze target estimation and social gaze prediction. In this paper, we explore the use of multimodal LLMs for gaze target estimation and show that they can be seamlessly extended to semantic and social gaze prediction tasks as well.

Vision-Language Models (VLMs) like LLaVA [40] and the Qwen-VL series [41–43] excel at generalized visual reasoning and have been successfully adapted for dense downstream tasks like visual grounding [44] and segmentation [18, 19, 45]. However, their direct application to gaze following remains limited. Prior works primarily utilize VLMs for auxiliary purposes, such as extracting zero-shot contextual embeddings used for existing vision-based models [46] or generating pseudo-annotations [47], rather than for direct localization. Concurrent VLM-based gaze models also exhibit notable limitations: GazeVLM [21] relies on auxiliary depth inputs and lags behind specialized spatial models in precision, while VL4Gaze [22] formulates the problem purely as VQA rather than evaluating on standard spatial localization benchmarks. Beyond generative VLMs, contrastive vision-language models such as CLIP [48] have also been leveraged for gaze-related tasks: CLIP-Gaze [49] uses language-described gaze-irrelevant factors to learn domain-generalizable features for gaze direction estimation, and VL-HTR [50] introduces language-guided query alignment and gaze direction priors from pretrained VLMs for gaze target estimation and gaze object prediction. In contrast, our work directly adapts VLMs for standard gaze target estimation, demonstrating state-of-the-art effectiveness across continuous spatial localization, semantic prediction, and complex social gaze reasoning.

3 Method

Given an input image $I \in \mathbb{R}^{H\times W\times 3}$ and a set of N head bounding boxes $\mathcal{B} = \{b_i\}_{i=1}^{N}$, where $b_i \in \mathbb{R}^4$ denotes the head region of person P_i, our primary objective is to predict each person's gaze status (whether the target is inside or outside the frame) and to estimate the precise target coordinates if it falls within the image (*where*). Furthermore, our framework seamlessly extends to higher-level relational tasks. For Semantic Gaze Reasoning, we infer the categorical identity of the object being attended to by each individual (*what*). For Social Gaze Interaction, we analyze every pair of individuals (P_i, P_j) to classify their relational dynamics into three distinct categories: Looking At Humans (LAH), Looking At Each Other (LAEO), and Shared Attention (SA).

3.1 Overview

Our framework is built upon a LoRA [51]-tuned Qwen3-VL [43] backbone that processes the image, system instruction prompt (see Appendix A), and structured multi-person prompts as input, where visual head embeddings are dynamically injected into the token sequence to ground each subject (Sec. 3.2). The architecture features a Structured Language Branch (Sec. 3.3), which leverages the VLM's discrete reasoning capacity to generate per-person textual predictions $\hat{T}_i$ including gaze status, coordinates, and semantic categories, and a Gaze Branch (Sec. 3.4), which decodes dense spatial signals from the backbone's hidden states to output high-resolution gaze heatmaps $M_i \in \mathbb{R}^{H'\times W'}$, along with binary in/out logits p_i^{in}, and pairwise social gaze logits $S_{i,j} \in \mathbb{R}^3$ from different prediction heads. By tapping into subject-specific hidden states and supervised with dense spatial heatmaps, this branch provides notably better localization than text-based coordinate generation. Therefore, we utilize the target derived from the dense heatmap M_i as our final spatial gaze prediction. Fig. 1 presents an overview of our framework.

3.2 Multimodal Input Construction and Gaze Token Injection

Given an input image I and a set of N head bounding boxes $\mathcal{B} = \{b_i\}_{i=1}^{N}$, where $b_i = (x_i^1, y_i^1, x_i^2, y_i^2)$, we first construct a structured text prompt by defining a dedicated slot T_i for each person using JSON format: `P`$_i$`:{"bbox_2d"` : $[x_i^1, y_i^1, x_i^2, y_i^2]\}$. In a standard VLM pipeline, the image I is patchified and encoded by a vision encoder into a sequence of global vision tokens $H_{vis} \in \mathbb{R}^{V\times D}$, where V is the number of image patches and D is the hidden dimension. Simultaneously, the system instructions and the person prompts $\mathcal{T} = \{T_i\}_{i=1}^{N}$ are tokenized and embedded into text tokens H_{text}. These

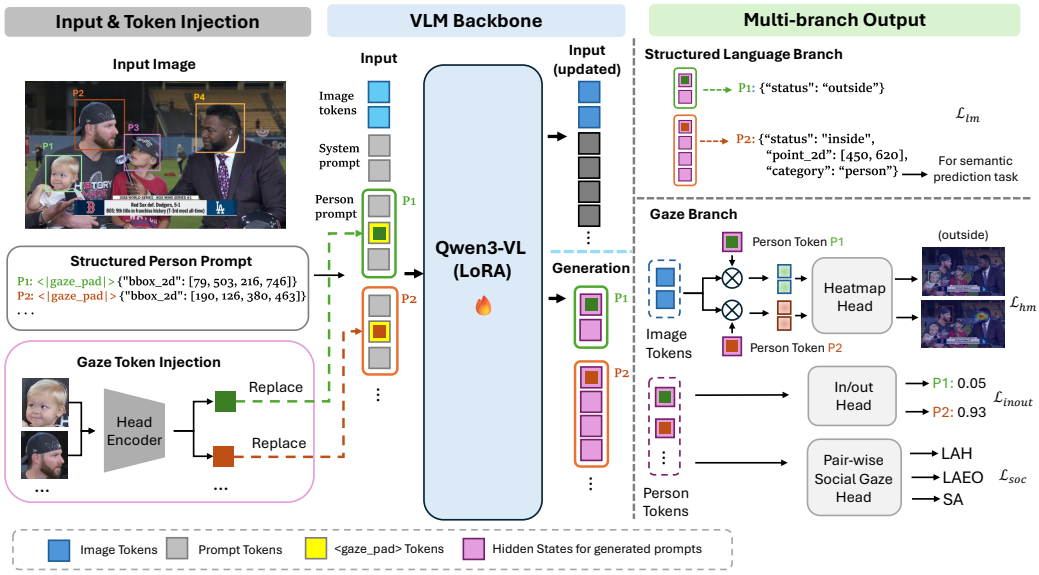

Figure 1: **Overall framework**. Given an image with multiple head bounding boxes, we construct a structured person prompt that assigns each person a <|gaze_pad|> placeholder followed by their head bounding box coordinates. The image, system prompt, and person prompt are processed by a LoRA-tuned Qwen3-VL. During token encoding, each <|gaze_pad|> embedding is replaced with the corresponding head embedding extracted from the cropped head, thereby injecting person-specific visual cues into the VLM. The structured language branch generates JSON outputs for gaze status, target coordinates, and optional semantic categories. The gaze branch extracts image tokens and person tokens from the generated hidden states, where each person token is anchored to the first token of its corresponding generated output sequence. The image and person tokens are combined to predict per-person gaze heatmaps, while the person tokens are further used for in/out prediction and pair-wise social gaze reasoning. For clarity, only two example persons are shown.

modalities are concatenated to form the initial sequence of hidden states $H^{(0)} \in \mathbb{R}^{L \times D}$ (where L is the total sequence length) before being fed into the VLM transformer blocks.

However, relying solely on this standard formulation presents two challenges for gaze target estimation. First, it forces the VLM to perform complex spatial referencing entirely through discrete text tokens. Second, the global vision tokens H_{vis} may lack the fine-grained, face-centric appearance and 3D orientation cues necessary to disambiguate gaze directions in complex scenes. To resolve this, we introduce a *head-conditioned gaze token injection* mechanism. We augment the input text by prepending a special placeholder token, <|gaze_pad|>, to each person's prompt entry T_i:

$$\texttt{P}_i\texttt{:<|gaze_pad|>}\ \{\texttt{"bbox_2d"} : [x_i^1, y_i^1, x_i^2, y_i^2]\}. \tag{1}$$

For each specified box b_i, we crop the corresponding head region c_i from the input image I and process it through a head encoder E_{head} to extract a gaze-aware visual embedding $e_i \in \mathbb{R}^D$. In our implementation, E_{head} utilizes a ResNet-18 backbone [52] pretrained on the Gaze360 dataset [53].

Let g_i denote the sequence index of the <|gaze_pad|> token associated with person P_i. Prior to the first transformer layer of the LLM, we replace the text embedding at this index $H_{g_i}^{(0)}$ with our extracted head embedding e_i. This targeted substitution explicitly injects localized head appearance and orientation cues into the sequence, grounding the person-specific prompt with rich visual features without disrupting the VLM's structured multimodal interface.

3.3 Structured Language Branch

The structured language branch utilizes the standard causal language modeling head of the VLM to autoregressively generate gaze predictions in a parseable JSON format. For each person P_i specified in the multimodal prompt, the model predicts their gaze status (i.e., whether their gaze target falls

inside or outside the image frame). If the target is inside the image, the model outputs the status alongside the normalized 2D target coordinates. For semantic gaze prediction tasks, the model optionally generates a text category describing the gazed-at object.

The generated text $\hat{T}_i$ for person P_i follows a strict schema:

$$\texttt{P}_i\texttt{:}\{\texttt{"status"}:\texttt{"inside"},\ \texttt{"point_2d"}:\ [x, y],\ \texttt{"category"}:\ \texttt{"<object>"}\} \tag{2}$$

Conversely, for targets located outside the image frame, the model simplifies the output schema to a single key-value pair: $\{\texttt{"status"}:\texttt{"outside"}\}$, omitting any further spatial or semantic attributes.

This text generation provides the discrete inference of the in/out status and the semantic category. Furthermore, the token sequence generated by this branch serves a critical structural purpose: the initial token of the output for each subject (e.g., the "P" token in "$\texttt{P}_i\texttt{:}$") acts as a spatial anchor. The hidden states at these specific positions provide person-specific embeddings rich in gaze-aware representations. As detailed in the following section, the gaze branch utilizes these specific token positions to extract person-aligned hidden states for dense geometric prediction. Here, the language branch primarily serves to enforce structured semantic reasoning and establish person-specific token anchors. We supervise the entire JSON text sequence using a standard autoregressive cross-entropy loss $\mathcal{L}_{lm}$. However, the 2D gaze target values generated within the text sequence are bypassed in favor of the gaze branch's dense predictions for the final localization result.

3.4 Gaze Branch via Hidden State Extraction

While the structured language branch outputs discrete text, continuous geometric and relational predictions are handled by the gaze branch using the model's internal representations. Specifically, we extract features from the hidden states of the final transformer layer in the VLM, denoted as $H^{(out)} \in \mathbb{R}^{L' \times D}$, where L' corresponds to the total length of the input and generated tokens.

To perform person-specific spatial reasoning, we isolate the hidden state corresponding to the P token in the generated $\texttt{P}_i\texttt{:}$ for each person as representative person embedding tokens. Because the VLM is instructed to generate personalized gaze information immediately following this token, the hidden state at this specific token position inherently aggregates and encodes rich, gaze-aware features for that individual. We denote this extracted person token as $h_{p_i} \in \mathbb{R}^{1 \times D}$. Simultaneously, we extract the sequence of updated image tokens $H'_{vis} \in \mathbb{R}^{V \times D}$ from $H^{(out)}$ to serve as the spatial canvas.

These extracted representations are then routed to specialized multi-task heads:

• **Gaze Target Estimation:** To predict the spatial target, the person token h_{p_i} and the updated image tokens H'_{vis} are fused via element-wise multiplication. This creates a personalized gaze feature for each person, which is then processed by a heatmap decoder consisting of several consecutive transposed convolutional layers to yield a high-resolution gaze target heatmap $M_i \in \mathbb{R}^{H' \times W'}$ for person i. This prediction is supervised by the Heatmap Loss $\mathcal{L}_{hm}$, which computes the Mean Squared Error (MSE) against a ground-truth heatmap generated by placing a 2D Gaussian distribution at the annotated target coordinate.

• **In/Out Prediction:** The person token h_{p_i} is passed independently through a Multi-Layer Perceptron (MLP) to output a binary probability logit p_i^{in}, reinforcing the discrete inside/outside status generated by the language branch. This prediction is supervised by $\mathcal{L}_{inout}$, a Binary Cross-Entropy (BCE) loss computed against the ground-truth in/out label.

• **Social Gaze Reasoning:** For complex scene understanding, we evaluate pairwise relational interactions between individuals. For every pair of people i and j, we concatenate their respective person tokens to form a joint representation $[h_{p_i}, h_{p_j}] \in \mathbb{R}^{1 \times 2D}$. This concatenated feature is passed through a three-layer MLP to estimate independent probability logits $S_{i,j} \in \mathbb{R}^3$ for three social gaze categories: LAH, LAEO, and SA. We apply a BCE loss independently to each logit to compute the overall Social Gaze Loss $\mathcal{L}_{soc}$.

3.5 Training Objectives

The entire framework is trained end-to-end in a multi-task manner with LoRA fine-tuning. The overall training objective is defined as a weighted combination of the individual task losses:

$$\mathcal{L}_{total} = \lambda_{lm}\mathcal{L}_{lm} + \lambda_{hm}\mathcal{L}_{hm} + \lambda_{inout}\mathcal{L}_{inout} + \lambda_{soc}\mathcal{L}_{soc}, \tag{3}$$

where λ_{lm}, λ_{hm}, λ_{inout}, and λ_{soc} are hyperparameters used to balance the multi-task contributions.

4 Experiments

4.1 Datasets

We conduct experiments on multiple benchmark datasets, which are described below.

GazeFollow [11] is a large-scale image benchmark for gaze following. It comprises around 130K person instances across 122K images. The test set features 4,782 gaze instances, with each target labeled by roughly 10 independent annotators. Furthermore, we utilize the semantic extension introduced by Tafasca et al. [38] for the semantic prediction task, which augments the dataset with target object categories. This extension provides pseudo-labels over about 3,500 categories for training and a manually curated test set spanning a 346-class vocabulary for semantic evaluation.

VideoAttentionTarget [12] is a video gaze following dataset consisting of 1,331 clips sourced from 50 YouTube TV shows. It provides 164K annotated instances across 71K frames, notably introducing binary inside/outside frame labels for the gaze targets. The test split was constructed by holding out 10 complete TV shows, yielding 31,978 test annotations.

ChildPlay [29] is a video dataset focusing on the gaze behaviors of children during social interactions. Sourced from 95 YouTube videos, it yields 401 short clips containing approximately 258K instances across 120K frames. Beyond standard head bounding boxes and 2D target coordinates, instances are annotated with gaze state classes, including target inside, target outside, and gaze shifts.

GazeHOI [38] is a benchmark designed for simultaneous spatial localization and semantic classification. Rather than relying on automatically labeled annotations, GazeHOI was constructed by repurposing five established Human-Object Interaction (HOI) datasets, isolating specific interaction verbs where a person's gaze is directed at the interacted object. In total, it provides 55,995 gaze instances across 463 distinct object categories, and we evaluate on its test split of 4,934 (person, object) pairs.

VSGaze [39] is a composite dataset specifically designed for social gaze prediction, aggregating data from VideoAttentionTarget, ChildPlay, VideoCoAtt [54], and UCO-LAEO [55]. The authors unified these disparate sources through a cross-annotation pipeline. First, they generated continuous head track annotations by detecting missing heads and tracking them across frames. Second, they cross-annotated the tasks: social gaze datasets (VideoCoAtt, UCO-LAEO) were augmented with spatial gaze targets approximated at bounding box centers, while the gaze following datasets were extended with social gaze labels. The resulting unified dataset contains 714K annotated gaze points with 444k, 30k, 420k positive pair-wise social gaze labels for LAH, LAEO, and SA respectively.

4.2 Evaluation Metrics

For gaze target localization, we compute the L2 distance (**Dist.**) on a normalized 1×1 image plane, extracting the predicted coordinate from the peak heatmap activation. To account for multiple annotators in GazeFollow, we report both **Avg Dist.**, the distance to the average of the annotated points, and **Min Dist.**, the distance to the closest annotated point. On GazeFollow, we additionally report the heatmap **AUC**. Following [15, 56], we do not report AUC on the single-annotation video benchmarks (VideoAttentionTarget and ChildPlay), where the ground-truth heatmap is an artificial Gaussian centered on a single annotated point and AUC is therefore uninformative [15]. AUC on all benchmarks is reported in Appendix C for completeness. We additionally report the precision of looking at heads (**PLAH**) [15] on all benchmarks, where a prediction is counted as positive only if the predicted and annotated gaze points fall within the same head box. PLAH values of Gaze-LLE are computed by us using its official checkpoints and the evaluation protocol of Sharingan [15], while those of other methods are taken from [15]. Target inside/outside prediction is evaluated via **Average Precision (AP)**.

For semantic gaze prediction on GazeFollow, we report top-1 accuracy (**Accuracy@1**) and **MultiAccuracy@1**, which accounts for multiple labels that can be assigned to the same subject. Because our framework generates text for semantic labels rather than probability logits, we omit Accuracy@3. On GazeHOI, we evaluate target localization using **GazeAcc**, which measures whether the predicted point falls within the ground-truth object bounding box, along with **Accuracy@1** for recognition.

Table 1: Comparison with state-of-the-art methods on GazeFollow and VideoAttentionTarget. In the column of input modality, I is image, D is depth, P is pose, and E is eyes. Multi-person denotes support for multi-person inference in a single forward pass. AUC is reported only on GazeFollow, which has multiple annotations per instance; AUC on all benchmarks is reported in Appendix C. Best numbers are marked as bold while second-best are underlined.

Method	Input	Multi-person	GazeFollow				VideoAttentionTarget		
			Avg Dist.↓	Min Dist.↓	AUC↑	PLAH↑	Dist.↓	AP↑	PLAH↑
Recasens [11]	I	×	0.190	0.113	0.878	–	–	–	–
Chong [23]	I	×	0.187	0.112	0.896	–	0.171	0.712	–
Lian [24]	I	×	0.145	0.081	0.906	–	–	–	–
Chong [12]	I	×	0.137	0.077	0.921	–	0.134	0.853	–
Fang [13]	I+D+E	×	0.124	0.067	0.922	–	0.108	0.896	–
Jin [57]	I	✓	0.126	0.076	0.919	–	0.134	0.880	–
Bao [25]	I+D+P	×	0.122	–	0.928	–	0.120	0.869	–
Jin [28]	I+D+P	×	0.118	0.063	0.920	–	0.104	0.895	–
Gupta [14]	I+D+P	×	0.114	0.056	0.943	–	0.110	0.879	–
Miao [26]	I+D	×	0.123	0.065	0.934	–	0.109	<u>0.908</u>	–
Tafasca [29]	I+D	×	0.125	0.064	0.936	0.622	0.109	0.834	<u>0.752</u>
Sharingan [15]	I	✓	0.113	0.057	0.944	0.667	0.107	0.891	0.738
MTGS [39]	I	✓	0.116	0.059	0.929	–	0.105	0.869	–
Gaze-LLE (ViT-B) [16]	I	✓	0.104	0.045	<u>0.956</u>	0.665	0.107	0.897	0.722
Gaze-LLE (ViT-L) [16]	I	✓	<u>0.099</u>	<u>0.041</u>	**0.958**	<u>0.679</u>	0.103	0.903	0.738
GazeVLM [21]	I+D	✓	0.123	0.061	0.936	–	<u>0.102</u>	0.898	–
OmniGF	I	✓	**0.091**	**0.040**	0.941	**0.753**	**0.096**	**0.923**	**0.759**
Human	–	–	0.096	0.040	0.924	–	0.051	0.925	–

For social gaze prediction on VSGaze, we follow the standard pairwise evaluation protocol established by MTGS [39], reporting **F1** scores for LAH and LAEO, and **AP** for SA.

4.3 Implementation Details

Our framework utilizes the Qwen3-VL-4B-Instruct backbone, adapted via LoRA ($r = 16, \alpha = 32$) applied to the query and value projections of all 36 transformer layers of the language model. The head bounding box coordinates are normalized to the relative coordinate range $[0, 999]$ in the input prompt following the grounding convention of Qwen3-VL. Input resolutions are fixed at 448×448 for the global scene and 224×224 for cropped heads, and gaze heatmaps are predicted at 64×64 for each person. We use the AdamW optimizer with a batch size of 8 and two gradient accumulation steps (effective batch size 16). On GazeFollow, we train for 20 epochs with learning rates of 10^{-4} for the LoRA adapter and 2.5×10^{-4} for the head encoder and prediction heads, using one epoch of linear warmup followed by cosine annealing. The loss coefficients are set to $\lambda_{lm} = 1.0$, $\lambda_{hm} = 10.0$, and $\lambda_{soc} = 1.0$ (when applicable), and $\lambda_{inout} = 0.1$ when training with in/out supervision. As the standard GazeFollow annotations contain only in-frame gaze targets, the GazeFollow model in Table 1 is trained with $\lambda_{inout} = 0$. The models for VideoAttentionTarget, ChildPlay, VSGaze, and the semantic gaze prediction models are fine-tuned from the model pretrained on GazeFollow. The fine-tuning learning rates, schedules, and checkpoint selection for each task are detailed in Appendix B. Experiments are conducted on a single NVIDIA H100 (80 GB) or NVIDIA RTX PRO 6000 Blackwell (96 GB) GPU. Training times vary from 2 to 30 hours across datasets.

4.4 Comparison with State-of-the-Art

We compare our proposed framework against existing state-of-the-art gaze following methods on the GazeFollow and VideoAttentionTarget datasets in Table 1, and on the ChildPlay dataset in Table 2. Our framework establishes a new state-of-the-art across all distance-based target estimation and in/out prediction metrics on all datasets. On GazeFollow, our method achieves an Avg Dist. of 0.091 and a Min Dist. of 0.040, surpassing the reported human performance baseline for Avg Dist. (0.096), while remaining competitive in AUC (0.941). OmniGF also achieves the highest PLAH on all three datasets. We note that AUC has been shown to favor heatmaps with multiple modes over precise localization of

Table 2: Comparison Results on ChildPlay.

Method	Dist.↓	AP↑	PLAH↑
Gupta [14]	0.113	0.983	–
Tafasca [29]	0.107	0.986	0.590
Sharingan [15]	0.106	0.990	0.600
MTGS [39]	0.117	0.993	–
Gaze-LLE (ViT-B) [16]	0.106	0.994	0.579
Gaze-LLE (ViT-L) [16]	0.101	0.994	0.612
OmniGF	**0.090**	**0.996**	**0.662**

Table 3: Ablation Results on GazeFollow.

Components			Metrics		
Language (LoRA)	Spatial Decoding	Token Inject.	Avg ↓	Min ↓	AUC ↑
			0.340	0.258	–
✓			0.114	0.055	–
	✓		0.167	0.103	0.904
✓	✓		0.104	0.047	0.939
✓	✓	✓	**0.091**	**0.040**	**0.941**



Figure 2: Visualizations of model predictions of OmniGF on GazeFollow (1st and 2nd columns), VideoAttentionTarget (3rd column) and ChildPlay (4th column). The first row shows predicted gaze target locations for multiple persons in an image. The 2nd row shows the predicted heatmap for a representative person in the image. Ground truth annotations are visualized in yellow, and the predicted point is visualized in red.

the gaze target [47, 34]. We therefore regard the distance metrics, which directly measure geometric localization error, as the primary indicator of performance. On VideoAttentionTarget, our model achieves the lowest spatial error (Dist. 0.096) and highly competitive in/out prediction performance (AP 0.923), closing the gap toward human-level recognition. Crucially, our model achieves these results by processing all people with a single VLM input using solely RGB image inputs. This significantly outperforms complex architectures that rely on auxiliary modalities (e.g., depth, pose) and establishes a clear performance margin over prior multi-person capable models. Furthermore, our model significantly outperforms GazeVLM, a concurrent VLM-based baseline that requires depth images as an additional input modality. Meanwhile, we validate our model's robustness on the ChildPlay dataset, where it achieves a new state-of-the-art Dist. of 0.090 and an AP of 0.996, outperforming recent frameworks like Sharingan and Gaze-LLE. Qualitative results in Fig. 2 show high-quality multi-person gaze predictions from our model across multiple datasets.

4.5 Ablation Study

To validate our architectural innovations, we conduct a comprehensive ablation study on the GazeFollow dataset (Table 3). We first evaluate the pretrained Qwen3-VL [43] backbone in a zero-shot setting. Using our exact system and person prompts to instruct the model to generate gaze status and target coordinates in text, it yields very poor localization (0.340 Avg Dist.). This confirms that off-the-shelf text generation struggles with precise spatial localization for gaze following. Fine-tuning the structured language branch via LoRA significantly reduces this error (0.114) but remains limited by the precision bottleneck of text-based target localization. Conversely, training only the continuous gaze branch for dense heatmap decoding proves suboptimal (0.167). Because the heatmap head relies on person token embeddings extracted from the prompt, failing to simultaneously fine-tune the language branch deprives these embeddings of the necessary gaze-related context for accurate heatmap prediction. However, combining both approaches into our dual-branch architecture overcomes these limitations (0.104), proving that the hidden state tokens produced during language generation serve as essential structural anchors for extracting high-resolution spatial heatmaps. Finally, integrating dynamic gaze token injection achieves our state-of-the-art performance (0.091 Avg Dist.). By inject-

Table 4: Cross-dataset evaluation. All models are trained on GazeFollow and evaluated on unseen datasets without fine-tuning. PLAH is not computed on GOO-Real as it does not contain human heads as gaze targets.

Method	VideoAttentionTarget		ChildPlay		GOO-Real
	Dist.↓	PLAH↑	Dist.↓	PLAH↑	Dist.↓
Chong [12]	0.119	–	0.121	–	0.334
Jin [28]	0.104	–	–	–	–
Tonini w/ UDA [59]	–	–	–	–	0.238
Miao [26]	0.109	–	0.113	–	0.202
Gupta [14]	0.137	–	0.142	–	–
Tafasca [29]	0.123	–	0.115	–	–
Gaze-LLE (ViT-B) [16]	0.105	0.713	0.114	0.505	<u>0.174</u>
Gaze-LLE (ViT-L) [16]	<u>0.100</u>	<u>0.743</u>	<u>0.101</u>	<u>0.582</u>	0.175
OmniGF	**0.098**	**0.772**	**0.094**	**0.633**	**0.166**

ing cropped visual head embeddings directly into the input sequence, this mechanism provides the explicit, fine-grained appearance and orientation cues necessary to accurately disambiguate targets. Further ablations, zero-shot results of other VLMs, a runtime analysis, and qualitative results of semantic and social gaze prediction are provided in Appendices D–I.

4.6 Cross-dataset Evaluation

To assess generalization to unseen data, we follow the zero-shot protocol of prior work [16]: we train OmniGF on GazeFollow and evaluate the model without finetuning on VideoAttentionTarget, ChildPlay, and GOO-Real [58]. As shown in Table 4, OmniGF achieves the lowest Dist. on all three unseen datasets and the highest PLAH on VideoAttentionTarget and ChildPlay. Consistent with our in-domain evaluation, AUC for this setting is reported in Appendix C.

4.7 Performance on Semantic Gaze Prediction

Beyond standard spatial coordinate estimation, our framework seamlessly extends to semantic gaze target recognition by generating an additional "category" field within the structured language output. Because our model generates open-vocabulary text rather than discrete class logits, predicted labels may not perfectly align with the predefined test vocabulary. To resolve this, we encode both our text predictions and the evaluation classes using CLIP (utilizing the prompt template "a photo of a [label]") and retrieve the ten most similar test classes by cosine similarity. If any of these candidates is a WordNet hypernym of the predicted label, we assign the one closest to it in the hypernym hierarchy; otherwise, we assign the class with the highest similarity.

Furthermore, we identified noticeable noise in the original pseudo-labeled GazeFollow training set established by Semgaze (e.g., a "cellphone" being held was incorrectly labeled as "person"). To ensure high-quality supervision, we refined these pseudo-labels and subsequently trained both our model and a corrected Semgaze* baseline on this improved data. We will publicly release these refined annotations to support future research. Conversely, for GazeHOI, we maintained the original semantic labels because these are derived directly from the ground-truth object bounding boxes of established Human-Object Interaction datasets.

As shown in Tables 5 and 6, OmniGF establishes new state-of-the-art performance on both datasets. On GazeFollow, it achieves a remarkable Acc@1 of 0.649, with a 23.1% improvement over the corrected SemGaze baseline. Besides, it also excels in both localization and recognition on GazeHOI, outperforming SemGaze by 23.8%. This confirms that our dual-branch VLM architecture effectively bridges the gap between precise spatial localization and high-level semantic reasoning.

4.8 Performance on Social Gaze Prediction

Finally, we extend our framework to the complex task of social gaze prediction using the VSGaze dataset. A unique challenge of this dataset is that it includes additional head bounding boxes and tracks that lack ground-truth gaze target annotations. To effectively train our structured language branch in

Table 5: Semantic gaze prediction results on GazeFollow. Semgaze* denotes the baseline trained on our refined ground-truth labels for a fair comparison.

Method	Localization		Recognition	
	Avg. Dist. ↓	Min. Dist. ↓	Acc@1 ↑	MultiAcc@1 ↑
Semgaze [38]	0.108	0.051	0.447	0.516
Semgaze* [38]	0.109	0.053	0.527	0.598
OmniGF	**0.092**	**0.040**	**0.649**	**0.739**

Table 6: Semantic gaze prediction on GazeHOI. Semgaze* is omitted as semantic labels for GazeHOI are derived directly from existing HOI datasets.

Method	Localization	Recognition
	GazeAcc ↑	Acc@1 ↑
Semgaze [38]	0.723	0.583
OmniGF	**0.786**	**0.722**

this dense multi-person environment, we introduce a selective loss masking strategy. Specifically, we input dummy gaze slots for these unannotated individuals and assign them a categorical gaze status of "unknown." During backpropagation, we dynamically exclude the tokens corresponding to these individuals from the language modeling gradient. This ensures the model learns to process all individuals in the scene simultaneously for global context, while focusing its parameter updates strictly on valid, annotated targets.

As shown in Table 7, our framework establishes a new state-of-the-art across all gaze and social reasoning metrics. Crucially, our model performs end-to-end (E2E) social gaze reasoning jointly with gaze localization, eliminating the need for the manual post-processing required by ordinary gaze following methods. Compared to the only E2E baseline MTGS [39], our model improves from 0.590 to 0.742 (25.8%) on F1_{LAEO} and from 0.576 to 0.787 (36.6%) on AP_{SA}. These unprecedented gains demonstrate that leveraging the semantic reasoning capabilities of a VLM provides a fundamental advantage in understanding complex social interactions over traditional spatial-only architectures.

Table 7: Comparison of social gaze reasoning and gaze target estimation performance on VSGaze. E2E denotes end-to-end prediction without relying on manual post-processing heuristics.

Method	E2E	Dist. ↓	AP_{IO} ↑	F1_{LAH} ↑	F1_{LAEO} ↑	AP_{SA} ↑
Chong_S [12]	×	0.121	0.918	0.778	0.562	0.288
Chong_T [12]	×	0.130	0.956	0.764	0.529	0.331
Gupta [14]	×	0.119	0.929	0.784	0.590	0.335
MTGS [39]	✓	0.107	0.940	0.795	0.590	0.576
OmniGF	✓	**0.098**	**0.992**	**0.814**	**0.742**	**0.787**

5 Conclusion

In this work, we presented OmniGF, a novel Vision-Language framework for unified gaze following. By introducing a dual-branch architecture that harmonizes a structured language branch with a continuous dense spatial decoder, alongside a dynamic gaze token injection mechanism, we successfully adapted a foundational VLM for high-precision spatial reasoning tasks. OmniGF establishes a new state-of-the-art across diverse benchmarks in a single multi-person forward pass. To the best of our knowledge, this is the first successful application of VLMs to gaze localization that outperforms state-of-the-art vision-based models. Furthermore, benefiting from VLM's strong semantic reasoning capacity, we demonstrate that the model can be seamlessly extended to semantic and social gaze prediction tasks, outperforming current state-of-the-art models by a significant margin. This demonstrates that foundational VLMs can fundamentally unify spatial localization and discrete high-level semantic reasoning. We hope that OmniGF provides a robust, streamlined foundation to accelerate future research in human behavior understanding, social scene analysis, and interactive AI systems.

Acknowledgements

This project was partially supported by NSF awards IIS-2123920 and IIS-2212046. Minh Hoai was initially supported by NSF award DUE-2055406, and later in part by the Australian Institute for Machine Learning (Adelaide University) and the Centre for Augmented Reasoning, an initiative of the Australian Government's Department of Education. The authors thank Xiang Li and Jingwei Zhang for helpful discussions.

References

[1] S. Sheikhi and J.-M. Odobez, "Combining dynamic head pose–gaze mapping with the robot conversational state for attention recognition in human–robot interactions," *Pattern Recognition Letters*, vol. 66, pp. 81–90, 2015.

[2] H. Admoni and B. Scassellati, "Social eye gaze in human-robot interaction: a review," *Journal of Human-Robot Interaction*, vol. 6, no. 1, pp. 25–63, 2017.

[3] A. Saran, S. Majumdar, E. S. Short, A. Thomaz, and S. Niekum, "Human gaze following for human-robot interaction," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 8615–8621, IEEE, 2018.

[4] I. Kasahara, S. Stent, and H. S. Park, "Look both ways: Self-supervising driver gaze estimation and road scene saliency," in *European Conference on Computer Vision*, pp. 126–142, Springer, 2022.

[5] S. Martin, S. Vora, K. Yuen, and M. M. Trivedi, "Dynamics of driver's gaze: Explorations in behavior modeling and maneuver prediction," *IEEE Transactions on Intelligent Vehicles*, vol. 3, no. 2, pp. 141–150, 2018.

[6] C. Liu, Y. Chen, L. Tai, H. Ye, M. Liu, and B. E. Shi, "A gaze model improves autonomous driving," in *Proceedings of the 11th ACM Symposium on Eye Tracking Research & Applications*, pp. 1–5, 2019.

[7] N. J. Emery, "The eyes have it: the neuroethology, function and evolution of social gaze," *Neuroscience & Biobehavioral Reviews*, vol. 24, no. 6, pp. 581–604, 2000.

[8] B. Massé, S. Ba, and R. Horaud, "Tracking gaze and visual focus of attention of people involved in social interaction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 11, pp. 2711–2724, 2018.

[9] A. Senju and M. H. Johnson, "Atypical eye contact in autism: Models, mechanisms and development," *Neuroscience & Biobehavioral Reviews*, vol. 33, no. 8, pp. 1204–1214, 2009.

[10] R. A. J. De Belen, T. Bednarz, A. Sowmya, and D. Del Favero, "Computer vision in autism spectrum disorder research: a systematic review of published studies from 2009 to 2019," *Translational Psychiatry*, vol. 10, no. 1, p. 333, 2020.

[11] A. Recasens, A. Khosla, C. Vondrick, and A. Torralba, "Where are they looking?," in *Advances in Neural Information Processing Systems* (C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, eds.), vol. 28, Curran Associates, Inc., 2015.

[12] E. Chong, Y. Wang, N. Ruiz, and J. M. Rehg, "Detecting attended visual targets in video," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5396–5406, 2020.

[13] Y. Fang, J. Tang, W. Shen, W. Shen, X. Gu, L. Song, and G. Zhai, "Dual attention guided gaze target detection in the wild," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11390–11399, 2021.

[14] A. Gupta, S. Tafasca, and J.-M. Odobez, "A modular multimodal architecture for gaze target prediction: Application to privacy-sensitive settings," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 5041–5050, 2022.

[15] S. Tafasca, A. Gupta, and J.-M. Odobez, "Sharingan: A transformer architecture for multi-person gaze following," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2008–2017, 2024.

[16] F. Ryan, A. Bati, S. Lee, D. Bolya, J. Hoffman, and J. M. Rehg, "Gaze-LLE: Gaze target estimation via large-scale learned encoders," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 28874–28884, 2025.

[17] K. Chen, Z. Zhang, W. Zeng, R. Zhang, F. Zhu, and R. Zhao, "Shikra: Unleashing multimodal LLM's referential dialogue magic," *arXiv preprint arXiv:2306.15195*, 2023.

[18] L. Beyer, A. Steiner, A. S. Pinto, A. Kolesnikov, X. Wang, D. Salz, M. Neumann, I. Alabdulmohsin, M. Tschannen, E. Bugliarello, T. Unterthiner, D. Keysers, S. Koppula, F. Liu, A. Grycner, A. Gritsenko, N. Houlsby, M. Kumar, K. Rong, J. Eisenschlos, R. Kabra, M. Bauer, M. Bošnjak, X. Chen, M. Minderer, P. Voigtlaender, I. Bica, I. Balazevic, J. Puigcerver, P. Papalampidi, O. Henaff, X. Xiong, R. Soricut, J. Harmsen, and X. Zhai, "PaliGemma: A versatile 3B VLM for transfer," *arXiv preprint arXiv:2407.07726*, 2024.

[19] L. H. Li, P. Zhang, H. Zhang, J. Yang, C. Li, Y. Zhong, L. Wang, L. Yuan, L. Zhang, J.-N. Hwang, K.-W. Chang, and J. Gao, "Grounded language-image pre-training," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[20] H. Wang, A. Gupta, P. Vuillecard, and J.-M. Odobez, "Eyes on VLM: Benchmarking gaze following and social gaze prediction in vision language models," *arXiv preprint arXiv:2605.19859*, 2026.

[21] A. M. Mathew, H. Hermassi, T. Khalid, and A. A. Khan, "GazeVLM: A vision-language model for multi-task gaze understanding," *arXiv preprint arXiv:2511.06348*, 2025.

[22] S. Wang, C. Cui, Y. Huang, H. J. Chang, and Y. Cheng, "VL4Gaze: Unleashing vision-language models for gaze following," *arXiv preprint arXiv:2512.20735*, 2025.

[23] E. Chong, N. Ruiz, Y. Wang, Y. Zhang, A. Rozga, and J. M. Rehg, "Connecting gaze, scene, and attention: Generalized attention estimation via joint modeling of gaze and scene saliency," in *Proceedings of the European Conference on Computer Vision*, pp. 383–398, 2018.

[24] D. Lian, Z. Yu, and S. Gao, "Believe it or not, we know what you are looking at!," in *Proceedings of the Asian Conference on Computer Vision*, pp. 35–50, Springer, 2018.

[25] J. Bao, B. Liu, and J. Yu, "ESCNet: Gaze target detection with the understanding of 3D scenes," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14126–14135, June 2022.

[26] Q. Miao, M. Hoai, and D. Samaras, "Patch-level gaze distribution prediction for gaze following," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2023.

[27] Z. Hu, D. Yang, S. Cheng, L. Zhou, S. Wu, and J. Liu, "We know where they are looking at from the RGB-D camera: Gaze following in 3D," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1–14, 2022.

[28] T. Jin, Q. Yu, S. Zhu, Z. Lin, J. Ren, Y. Zhou, and W. Song, "Depth-aware gaze-following via auxiliary networks for robotics," *Engineering Applications of Artificial Intelligence*, vol. 113, p. 104924, 2022.

[29] S. Tafasca, A. Gupta, and J.-M. Odobez, "ChildPlay: A new benchmark for understanding children's gaze behaviour," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 20935–20946, 2023.

[30] Y. Yang, Y. Yin, and F. Lu, "Gaze target detection by merging human attention and activity cues," in *Proceedings of AAAI Conference on Artificial Intelligence*, pp. 6585–6593, 2024.

[31] R. Bachmann, D. Mizrahi, A. Atanov, and A. Zamir, "MultiMAE: Multi-modal multi-task masked autoencoders," in *Proceedings of the European Conference on Computer Vision*, pp. 348–367, Springer, 2022.

[32] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, *et al.*, "DINOv2: Learning robust visual features without supervision," *Transactions on Machine Learning Research*, 2024.

[33] D. Tu, X. Min, H. Duan, G. Guo, G. Zhai, and W. Shen, "End-to-end human-gaze-target detection with transformers," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2192–2200, IEEE, 2022.

[34] F. Tonini, N. Dall'Asen, C. Beyan, and E. Ricci, "Object-aware gaze target detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 21860–21869, 2023.

[35] D. Tu, W. Shen, W. Sun, X. Min, and G. Zhai, "Joint gaze-location and gaze-object detection," *arXiv preprint arXiv:2308.13857*, 2023.

[36] Y. Hou, Z. Zhang, N. Horanyi, J. Moon, Y. Cheng, and H. J. Chang, "Multi-modal gaze following in conversational scenarios," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 1186–1195, 2024.

[37] Q. Miao, V. R. Golani, J. Xu, P. P. Dutta, M. Hoai, and D. Samaras, "Multi-view gaze target estimation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025.

[38] S. Tafasca, A. Gupta, V. Bros, and J.-M. Odobez, "Toward semantic gaze target detection," in *Advances in Neural Information Processing Systems*, vol. 37, pp. 121422–121448, 2024.

[39] A. Gupta, S. Tafasca, A. Farkhondeh, P. Vuillecard, and J.-M. Odobez, "MTGS: A novel framework for multi-person temporal gaze following and social gaze prediction," in *Advances in Neural Information Processing Systems*, 2024.

[40] H. Liu, C. Li, Q. Wu, and Y. J. Lee, "Visual instruction tuning," in *Advances in Neural Information Processing Systems*, vol. 36, 2023.

[41] J. Bai, S. Bai, S. Yang, S. Wang, S. Tan, P. Wang, J. Lin, C. Zhou, and J. Zhou, "Qwen-VL: A versatile vision-language model for understanding, localization, text reading, and beyond," *arXiv preprint arXiv:2308.12966*, 2023.

[42] P. Wang, S. Bai, S. Tan, S. Wang, Z. Fan, J. Bai, K. Chen, X. Liu, J. Wang, W. Ge, *et al.*, "Qwen2-VL: Enhancing vision-language model's perception of the world at any resolution," *arXiv preprint arXiv:2409.12191*, 2024.

[43] S. Bai, Y. Cai, R. Chen, K. Chen, X. Chen, Z. Cheng, L. Deng, W. Ding, C. Gao, C. Ge, *et al.*, "Qwen3-VL technical report," *arXiv preprint arXiv:2511.21631*, 2025.

[44] H. Rasheed, M. Maaz, S. Shaji, A. Shaker, S. Khan, H. Cholakkal, R. M. Anwer, E. Xing, M.-H. Yang, and F. S. Khan, "GLaMM: Pixel grounding large multimodal model," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2024.

[45] X. Lai, Z. Tian, Y. Chen, Y. Li, Y. Yuan, S. Liu, and J. Jia, "LISA: Reasoning segmentation via large language model," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9579–9589, June 2024.

[46] A. Gupta, P. Vuillecard, A. Farkhondeh, and J.-M. Odobez, "Exploring the zero-shot capabilities of vision-language models for improving gaze following," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2024.

[47] Q. Miao, A. Graikos, J. Zhang, S. Mondal, M. Hoai, and D. Samaras, "Diffusion-refined VQA annotations for semi-supervised gaze following," in *European Conference on Computer Vision*, 2024.

[48] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," in *Proceedings of the International Conference on Machine Learning*, pp. 8748–8763, 2021.

[49] P. Yin, G. Zeng, J. Wang, and D. Xie, "CLIP-Gaze: Towards general gaze estimation via visual-linguistic model," in *Proceedings of AAAI Conference on Artificial Intelligence*, pp. 6729–6737, 2024.

[50] B. Wang, C. Guo, J. Cui, H. Xia, G. Guo, and Z. Li, "VL-HTR: Learning human–target representation from vision–language model," *IEEE Transactions on Cybernetics*, vol. 56, no. 5, pp. 2638–2649, 2026.

[51] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "LoRA: Low-rank adaptation of large language models," in *International Conference on Learning Representations*, 2022.

[52] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016.

[53] P. Kellnhofer, A. Recasens, S. Stent, W. Matusik, and A. Torralba, "Gaze360: Physically unconstrained gaze estimation in the wild," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6912–6921, 2019.

[54] L. Fan, Y. Chen, P. Wei, W. Wang, and S.-C. Zhu, "Inferring shared attention in social scene videos," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6460–6468, 2018.

[55] M. J. Marin-Jimenez, V. Kalogeiton, P. Medina-Suarez, and A. Zisserman, "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.

[56] X. Cao, H. Yang, V. Gunda, Z. Zhou, T. Xu, A. Kowdle, I. Kim, and J. M. Rehg, "Gaze target estimation anywhere with concepts," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026.

[57] T. Jin, Z. Lin, S. Zhu, W. Wang, and S. Hu, "Multi-person gaze-following with numerical coordinate regression," in *2021 16th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2021)*, pp. 1–8, IEEE, 2021.

[58] H. Tomas, M. Reyes, R. Dionido, M. Ty, J. Mirando, J. Casimiro, R. Atienza, and R. Guinto, "GOO: A dataset for gaze object prediction in retail environments," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 3119–3127, 2021.

[59] F. Tonini, C. Beyan, and E. Ricci, "Multimodal across domains gaze target detection," in *Proceedings of the 2022 International Conference on Multimodal Interaction*, pp. 420–431, 2022.

[60] W. Wang, Z. Gao, *et al.*, "InternVL3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency," *arXiv preprint arXiv:2508.18265*, 2025.

[61] Google DeepMind, "Gemini 3.6 Flash." `https://aistudio.google.com/docs/models/gemini-3.6-flash`, 2026.

A System Prompts

To effectively guide the Vision-Language Model (VLM) backbone to generate structured reasoning states, we utilize specific system prompts during both training and inference. These prompts inform the model of the input format, establish a normalized coordinate space, and strictly dictate the JSON schema of the desired output.

Basic Gaze Target Estimation. For standard spatial gaze target estimation, we use the following base system prompt:

> *"You are given an image with head bounding boxes for all people in the scene. Each person is identified as P0..PN with their head box given as* `{"bbox_2d": [x1, y1, x2, y2]}` *in coordinates [0, 999]. For each person, predict whether their gaze target is inside or outside the image. Output one JSON object per person. If the gaze target is inside, output* `{"status": "inside", "point_2d": [x, y]}`*. If outside, output* `{"status": "outside"}`*."*

Semantic Gaze Prediction. For the semantic gaze target recognition task (evaluated on GazeFollow and GazeHOI), we extend the base prompt to simultaneously reason for the gaze target semantics. We achieve this by appending an instruction to predict what the person is looking at and updating the required JSON schema for in-frame targets. The full system prompt is formulated as follows:

> *"You are given an image with head bounding boxes for all people in the scene. Each person is identified as P0..PN with their head box given as* `{"bbox_2d": [x1, y1, x2, y2]}` *in coordinates [0, 999]. For each person, predict whether their gaze target is inside or outside the image, and what they are looking at. Output one JSON object per person. If the gaze target is inside, output* `{"status": "inside", "point_2d": [x, y], "category": "<object>"}`*. If outside, output* `{"status": "outside"}`*."*

B Additional Implementation Details

All models use the Qwen3-VL-4B-Instruct backbone with LoRA ($r = 16$, $\alpha = 32$) applied to the query and value projections of all 36 transformer layers of the language model, a Gaze360-pretrained ResNet-18 head encoder [53], 448×448 scene inputs, 224×224 head crops, and 64×64 gaze heatmaps per person. All models are trained on a single GPU with AdamW, a batch size of 8, and two gradient accumulation steps (effective batch size 16), with loss coefficients $\lambda_{lm} = 1.0$ and $\lambda_{hm} = 10.0$, and $\lambda_{soc} = 1.0$ for social gaze prediction. One learning rate is used for the LoRA parameters and another for the head encoder and all prediction heads.

On GazeFollow, one epoch covers 103,455 training images, i.e., 6,466 optimizer steps, and 20 epochs amount to 129,320 steps. We train two GazeFollow models: the model reported in Table 1, trained on the standard annotations with $\lambda_{inout} = 0$, and a model trained with in/out supervision ($\lambda_{inout} = 0.1$) on training annotations that additionally include out-of-frame persons, which serves as the initialization for VideoAttentionTarget, ChildPlay, and VSGaze. For every task, we select the checkpoint with the best validation metric (Dist. for gaze localization, Acc@1 for semantic gaze prediction). On VideoAttentionTarget and ChildPlay, the best validation distance is reached after the first epoch, i.e., after 568 and 1,037 optimizer steps, respectively. Table 8 summarizes the per-task settings.

Table 8: Per-task training settings. LR is given as LoRA / head encoder and prediction heads. Cosine (T) denotes one epoch of linear warmup followed by cosine annealing over T epochs.

Task / dataset	LR	Schedule
GazeFollow	10^{-4} / 2.5×10^{-4}	Cosine (20)
VideoAttentionTarget	2.5×10^{-5} / 5×10^{-5}	Constant
ChildPlay	2.5×10^{-5} / 2.5×10^{-5}	Cosine (3)
Semantic, GazeFollow	5×10^{-5} / 10^{-4}	Cosine (15)
Semantic, GazeHOI	5×10^{-5} / 10^{-4}	Constant
Social, VSGaze	5×10^{-5} / 5×10^{-5}	Constant

C Additional Evaluation Metrics: AUC and PLAH

In the main paper, we report AUC only on GazeFollow. AUC has known limitations in gaze following evaluation. For single-annotation benchmarks such as VideoAttentionTarget and ChildPlay, the ground-truth heatmap is an artificially constructed Gaussian with a predefined radius centered on the single annotated gaze point, so AUC merely measures similarity to this synthetic target and is largely uninformative [15]. Consequently, several recent methods have phased out AUC in favor of distance-based metrics, either for single-annotation benchmarks [15, 56, 39] or for all gaze following benchmarks [30]. For completeness, we report AUC on all benchmarks, together with the looking-at-heads precision (PLAH) used by [15], in both the in-domain (Table 9) and cross-dataset (Table 10) settings. We follow the PLAH definition of [15], under which a prediction is counted as positive only if the predicted and annotated gaze points fall within the same head box. PLAH values for Tafasca et al. [29] and Sharingan are taken from [15]. PLAH for Gaze-LLE was computed by us with the same evaluation code as for OmniGF, using the official checkpoints released by the authors. As GOO-Real does not contain human heads as gaze targets, PLAH is not computed on it.

Although OmniGF does not achieve the highest AUC, it achieves the best distance and PLAH scores on all datasets in both settings. We maintain that distance metrics, which directly evaluate the geometric precision of the predicted gaze target, remain the most reliable indicator of localization performance for this task.

Table 9: In-domain comparison including AUC and PLAH on GazeFollow (GF), VideoAttentionTarget (VAT), and ChildPlay (CP). Avg and Min denote the Avg Dist. and Min Dist. on GazeFollow. Best numbers are marked as bold while second-best are underlined.

Method	GazeFollow				VideoAttentionTarget				ChildPlay			
	Avg↓	Min↓	AUC↑	PLAH↑	Dist.↓	AUC↑	AP↑	PLAH↑	Dist.↓	AUC↑	AP↑	PLAH↑
Chong [12]	0.137	0.077	0.921	–	0.134	0.860	0.853	–	–	–	–	–
Miao [26]	0.123	0.065	0.934	–	0.109	0.917	<u>0.908</u>	–	–	–	–	–
Gupta [14]	0.114	0.056	0.943	–	0.110	–	0.879	–	0.113	0.919	0.983	–
Tafasca [29]	0.125	0.064	0.936	0.622	0.109	–	0.834	<u>0.752</u>	0.107	0.935	0.986	0.590
Sharingan [15]	0.113	0.057	0.944	0.667	0.107	–	0.891	0.738	0.106	–	0.990	0.600
MTGS [39]	0.116	0.059	0.929	–	0.105	–	0.869	–	0.117	–	0.993	–
Gaze-LLE (ViT-B) [16]	0.104	0.045	<u>0.956</u>	0.665	0.107	0.933	0.897	0.722	0.106	<u>0.949</u>	<u>0.994</u>	0.579
Gaze-LLE (ViT-L) [16]	<u>0.099</u>	<u>0.041</u>	**0.958**	<u>0.679</u>	0.103	<u>0.937</u>	0.903	0.738	<u>0.101</u>	**0.951**	<u>0.994</u>	<u>0.612</u>
GazeVLM [21]	0.123	0.061	0.936	–	<u>0.102</u>	**0.948**	0.898	–	–	–	–	–
OmniGF (proposed)	**0.091**	**0.040**	0.941	**0.753**	**0.096**	0.917	**0.923**	**0.759**	**0.090**	0.939	**0.996**	**0.662**

Table 10: Cross-dataset evaluation including AUC and PLAH. All models are trained on GazeFollow and evaluated on unseen datasets without fine-tuning. PLAH is not computed on GOO-Real as it does not contain human heads as gaze targets.

Method	VideoAttentionTarget			ChildPlay			GOO-Real	
	Dist.↓	AUC↑	PLAH↑	Dist.↓	AUC↑	PLAH↑	Dist.↓	AUC↑
Chong [12]	0.119	0.906	–	0.121	0.912	–	0.334	0.670
Jin [28]	0.104	0.900	–	–	–	–	–	–
Tonini w/ UDA [59]	–	–	–	–	–	–	0.238	0.840
Miao [26]	0.109	0.923	–	0.113	0.933	–	0.202	0.869
Gupta [14]	0.137	0.907	–	0.142	0.923	–	–	–
Tafasca [29]	0.123	0.911	–	0.115	0.932	–	–	–
Gaze-LLE (ViT-B) [16]	0.105	<u>0.932</u>	0.713	0.114	<u>0.946</u>	0.505	<u>0.174</u>	**0.901**
Gaze-LLE (ViT-L) [16]	<u>0.100</u>	**0.937**	<u>0.743</u>	<u>0.101</u>	**0.951**	<u>0.582</u>	0.175	<u>0.898</u>
OmniGF (proposed)	**0.098**	0.908	**0.772**	**0.094**	0.933	**0.633**	**0.166**	0.861

D Ablation on Person Token Ordering

Since OmniGF processes all individuals in a scene jointly through a single structured prompt, the person tokens $(P_1, P_2, \ldots, P_N)$ must be presented to the model in some order. During training, we

randomly shuffle the person order. In all other experiments, persons are presented at inference time in their default order in the dataset annotation files, without enforcing any specific spatial or logical ordering. To assess the sensitivity of the model to this sequence, we evaluate several person-ordering strategies at inference time on VideoAttentionTarget and ChildPlay, which naturally contain multiple people per image.

As shown in Table 11, varying the person ordering results in only negligible performance fluctuations (at most 0.004 in Dist. and AP). This confirms that OmniGF does not rely on a specific sequence of person tokens and remains robust to the order in which individuals are presented.

Table 11: Sensitivity to person-prompt ordering at inference time. LR–TB and RL–BT denote left-to-right/top-to-bottom and right-to-left/bottom-to-top spatial ordering, respectively. Random denotes the average over 3 random permutations with different seeds. Default denotes the order in the dataset annotation files, used in all other experiments.

Person ordering	ChildPlay		VideoAttentionTarget	
	Dist.↓	AP↑	Dist.↓	AP↑
LR–TB	0.090	0.997	0.097	0.923
RL–BT	0.093	0.996	0.100	0.919
Random	0.090	0.996	0.098	0.919
Default	0.090	0.996	0.096	0.923

E Ablation on Architectural Design Choices

To complement the component-level ablation in the main paper (Table 3), we evaluate finer-grained design choices along three axes: the position from which the person token is extracted, the strategy for injecting head embeddings, and the backbone layer from which hidden states are taken for the gaze branch. For each experiment, we vary one component at a time while keeping the rest of the model and training configuration unchanged, and train and evaluate on GazeFollow.

As shown in Table 12, using the first generated token of each person's output as the person token significantly outperforms both mean-pooling over the person's generated tokens and using the last generated token. We hypothesize that the first token acts as an anchor that cleanly captures the initial reasoning state for that individual, before it is mixed with the information needed to generate the remaining text tokens. For head token injection, replacing the dedicated <|gaze_pad|> embedding with the head embedding yields better localization than adding the head embedding to the embedding of the person identifier token (P_i) in the input prompt. Regarding hidden layer selection, a multi-layer DPT-style fusion of four intermediate layers offers a marginal improvement in AUC, whereas taking the hidden states from the final layer provides the best Avg Dist. and Min Dist. without the computational overhead of fusing multiple high-resolution feature maps.

Table 12: Ablation on architectural design choices on GazeFollow. Each block varies one component while keeping the others fixed. Our final design is shaded.

Axis	Variant	Avg Dist.↓	Min Dist.↓	AUC↑
Person token	Last generated token	0.288	0.216	0.566
	Mean-pooled generated tokens	0.104	0.047	0.852
	First generated token	0.091	0.040	0.941
Head injection	Add to P_i token embedding	0.096	0.042	0.939
	Replace <|gaze_pad|> embedding	0.091	0.040	0.941
Hidden layer	Layer 9 (early)	0.103	0.044	0.940
	Layer 18 (middle)	0.095	0.041	0.943
	DPT-style fusion of 4 layers	0.094	0.042	0.944
	Final layer (36)	0.091	0.040	0.941

F Zero-shot Performance of Other VLMs

In the ablation study (Table 3), we report the zero-shot performance of the Qwen3-VL-4B-Instruct backbone. To examine whether this limitation persists with larger or different foundation models, we further evaluate Qwen3-VL at multiple scales (2B, 4B, 8B, and 32B), InternVL3.5-8B, and the commercial model Gemini-3.6-Flash on the GazeFollow test set in a zero-shot setting. All models are prompted with the same system and person prompts used by OmniGF (Appendix A) to generate the gaze status and target coordinates for every person in text. We use deterministic greedy decoding with a single returned sequence and parse the JSON entry associated with each person. Invalid or missing entries are counted as failed predictions and assigned the image center $[0.5, 0.5]$ as the prediction. Due to API cost constraints, Gemini-3.6-Flash is evaluated on a randomly sampled subset of 1.5K test instances.

As shown in Table 13, localization generally improves with model scale, yet even the strongest models remain far from task-specific performance: the best zero-shot result (0.231 Avg Dist. by Gemini-3.6-Flash) is more than twice the error of OmniGF (0.091). This indicates that, while large foundation models exhibit some zero-shot understanding of gaze, they still struggle with the precise continuous spatial localization required for gaze following, motivating the dual-branch design of OmniGF.

Table 13: Zero-shot gaze target estimation of VLMs on the GazeFollow test set. † Evaluated on a randomly sampled subset of 1.5K test instances due to API cost constraints. OmniGF is included for reference.

Model	Avg Dist.↓	Min Dist.↓
Qwen3-VL-2B-Instruct [43]	0.341	0.255
Qwen3-VL-4B-Instruct [43]	0.340	0.258
Qwen3-VL-8B-Instruct [43]	0.286	0.202
Qwen3-VL-32B-Instruct [43]	0.252	0.174
InternVL3.5-8B [60]	0.271	0.202
Gemini-3.6-Flash† [61]	0.231	0.165
OmniGF (Qwen3-VL-4B, fine-tuned)	0.091	0.040

G Ablation on Loss Coefficients

We study the sensitivity of OmniGF to the coefficients of the two primary loss terms in the training objective (Sec. 3.5): the language modeling loss weight λ_{lm} and the heatmap loss weight λ_{hm}. We vary one coefficient at a time while keeping all other settings fixed, and train on GazeFollow.

As shown in Table 14, the model is robust to moderate changes of λ_{lm}, with Avg Dist. varying between 0.091 and 0.095. The heatmap loss weight has a larger effect: a small λ_{hm} (0.1) noticeably degrades localization (0.102 Avg Dist.), as the dense spatial supervision is underweighted relative to the language modeling objective, while an overly large weight (100) also slightly hurts performance. Our default setting ($\lambda_{lm} = 1.0$, $\lambda_{hm} = 10.0$) achieves the best Avg Dist. and Min Dist. Variations in AUC across settings are small (within 0.008).

Table 14: Ablation of loss coefficients on GazeFollow. The default setting used in all other experiments is shaded. Best numbers are marked as bold.

λ_{lm}	λ_{hm}	Avg Dist.↓	Min Dist.↓	AUC↑
0.5	10.0	0.095	0.041	0.943
1.0	10.0	**0.091**	**0.040**	0.941
2.0	10.0	0.092	**0.040**	0.942
1.0	0.1	0.102	0.047	0.936
1.0	1.0	0.094	0.041	**0.944**
1.0	10.0	**0.091**	**0.040**	0.941
1.0	100.0	0.096	0.042	0.942

H Scalability and Runtime Analysis

A key property of OmniGF is that all individuals in a scene are processed jointly, so the scene is encoded once per image regardless of the number of people, rather than once per person as in most prior gaze following methods. To quantify the resulting scaling, we measure the inference time and computational cost (TFLOPs) of OmniGF for images with varying numbers of people N. To obtain a controlled curve up to $N = 8$, beyond the person counts available in most test frames, we attach N synthetic head boxes to a real image and fix the generation length to 25 tokens per person, the length of one in-frame prediction. Runtime is reported per image, measured while processing batches of 64 images on a single NVIDIA RTX PRO 6000 Blackwell GPU with bfloat16 mixed precision.

As shown in Table 15, when the number of people increases from 1 to 8, the computational cost increases by only $1.92\times$, rather than the $8\times$ incurred by running the backbone once per person, and the per-image runtime in this batched setting increases by $3.28\times$. Consequently, the per-person cost drops from 7.28 to 1.74 TFLOPs, and the per-person runtime from 53.62 ms to 22.02 ms. We note that the absolute inference cost of OmniGF remains higher than that of smaller task-specific models due to its 4B-parameter VLM backbone, although only 29.62M parameters are trainable through parameter-efficient fine-tuning. We discuss this limitation further in Appendix J.

Table 15: Inference runtime and computational cost of OmniGF with varying numbers of people N in an image, using synthetic head boxes and 25 generated tokens per person. Runtime is per image, measured with batches of 64 images. Per-person values are the totals divided by N.

	$N = 1$	$N = 2$	$N = 4$	$N = 8$
Runtime (ms)	53.62	71.14	102.59	176.14
Per-person runtime (ms)	53.62	35.57	25.65	22.02
TFLOPs	7.28	8.22	10.10	13.95
Per-person TFLOPs	7.28	4.11	2.53	1.74

I Qualitative Results on Auxiliary Tasks

In this section, we present qualitative visualizations across two extended gaze-following tasks: semantic gaze prediction and social gaze prediction.

In Fig. 3, we visualize the predicted gaze targets alongside their predicted semantic labels on the GazeFollow and GazeHOI datasets. As demonstrated, our model accurately localizes the spatial coordinate of the gaze target while simultaneously predicting highly context-aware semantic labels.

In Fig. 4, we illustrate the predicted social gaze labels across diverse interaction scenarios in the VSGaze dataset. The visualizations demonstrate the model's capacity to robustly predict multi-person social gaze dynamics and complex inter-subject gaze relationships across various challenging sub-datasets.

J Limitations and Future Work

While OmniGF establishes state-of-the-art performance across multiple benchmarks, we acknowledge a few limitations that provide avenues for future research. First, although our framework achieves single-pass multi-person inference, its reliance on a foundational VLM backbone inherently demands a larger memory footprint and computational budget compared to legacy CNN-based architectures, and its autoregressive decoding time grows with the number of people in the scene. Future deployments on edge devices could benefit from model quantization or distillation. Second, OmniGF is fundamentally an image-based architecture. When evaluated on video datasets (such as VideoAttentionTarget and ChildPlay), it processes scenes strictly on a frame-by-frame basis without explicit temporal modeling. While it cannot currently leverage motion cues or temporal smoothing to resolve highly ambiguous dynamic gaze behaviors, it is noteworthy that our frame-level approach still outperforms prior methods that explicitly incorporate temporal fusion. Future extensions could integrate temporal information to further boost robustness in video domains. Finally, while our dynamic gaze token injection scales elegantly to multi-person scenes, localization performance can

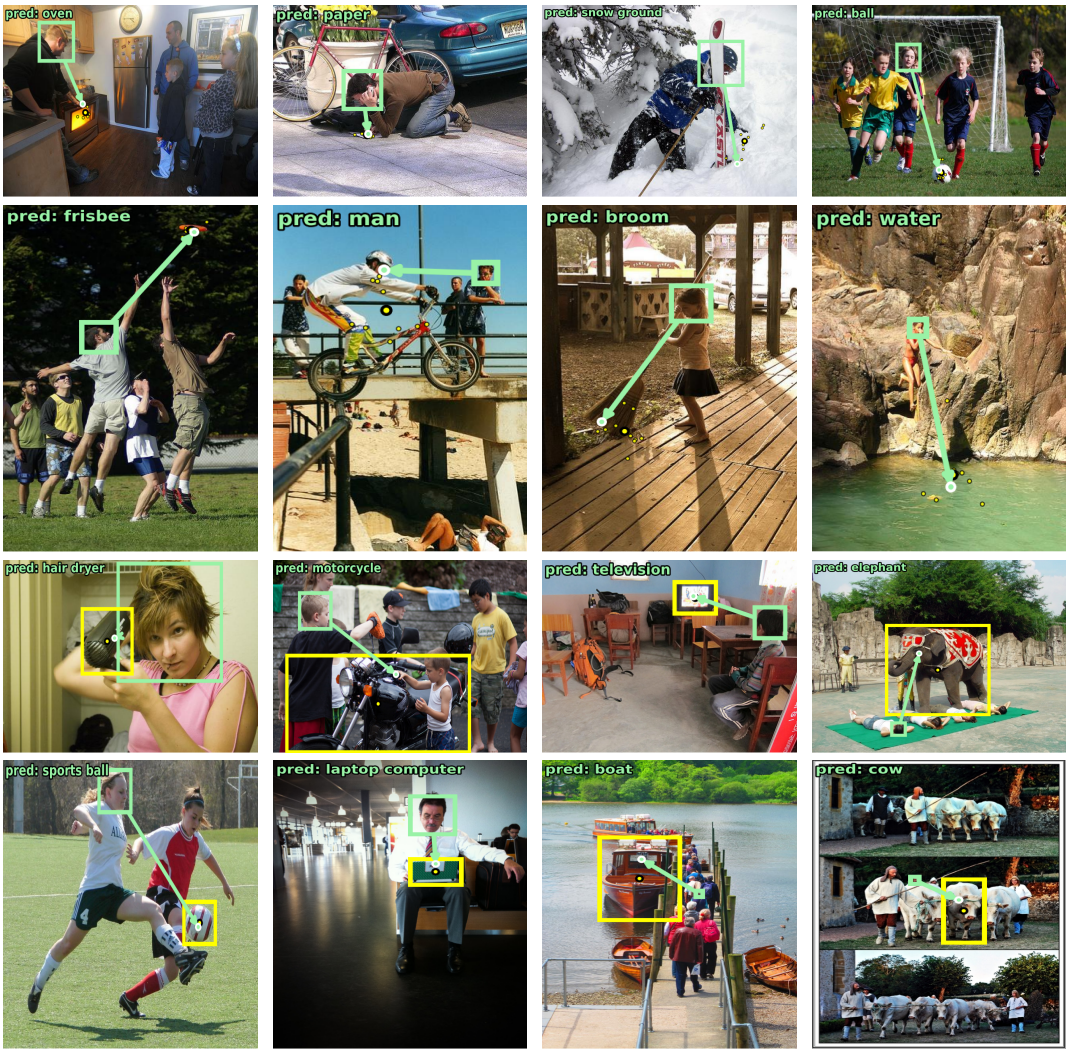

Figure 3: Visualizations of the model predictions with semantic gaze labels on GazeFollow (first two rows) and GazeHOI (last two rows). Yellow dots or bounding boxes indicate the ground truth gaze target. Note that the GazeHOI dataset marks the ground truth gaze target coordinate as the bounding box center of the gaze target object.

experience minor drops in extremely crowded environments (e.g., scenes with over 15 individuals in VSGaze). In such extreme cases, the extended input token sequence may lead to attention dilution. Nevertheless, because the scene is encoded only once per image, the computational cost of OmniGF grows much more slowly with the number of people than the person-by-person processing required by prior multi-branch architectures (Appendix H).

K Broader Impacts

Our proposed framework, OmniGF, advances the capabilities of Vision-Language Models in understanding human gaze and complex social interactions. On the positive side, high-precision gaze estimation holds significant potential for beneficial applications, such as developing more intuitive human-computer interfaces, enabling socially aware collaborative robotics, and providing assistive technologies for individuals with disabilities. Furthermore, it can serve as a valuable analytical tool for behavioral research in psychology and healthcare, potentially aiding in the study of neurodevelopmental conditions such as autism. However, we also acknowledge the potential negative societal impacts of this technology. The ability to accurately track and analyze where individuals



Figure 4: Qualitative results of social gaze predictions on the VSGaze dataset. Each row presents examples from a constituent sub-dataset: VideoAttentionTarget (Row 1), ChildPlay (Row 2), UCO-LAEO (Row 3), and VideoCoAtt (Row 4). Pairs engaging in Looking At Each Other (LAEO) are connected by solid blue lines, while individuals exhibiting Shared Attention (SA) are connected by dashed purple lines.

are looking in dense, multi-person scenes raises privacy concerns. If deployed irresponsibly, such systems could be misused for intrusive public surveillance, non-consensual tracking of consumer attention in retail environments, or pervasive employee monitoring. Additionally, because our method relies on foundational VLMs, it may inherit or amplify underlying biases present in the large-scale pre-training data, which could lead to disparate performance across different demographic groups. We encourage the community to establish strict ethical guidelines and prioritize privacy-preserving mechanisms when deploying gaze-aware systems in real-world environments.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction explicitly outline our core contributions: the OmniGF framework, head-conditioned token injection, and dual-branch decoding for unified multi-person gaze following. These claims accurately reflect the scope of the paper and are directly supported by the methodology and extensive experimental validation in the subsequent sections.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper includes a dedicated "Limitations and Future Work" section in Appendix J.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [N/A]

Justification: This paper focuses entirely on introducing a novel empirical architecture and demonstrating its performance through experimental validation. It does not propose or rely upon any theoretical results, lemmas, or mathematical proofs.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: Section 4.3 provides comprehensive implementation details (including hyperparameters and model settings), Appendix B lists the per-task training settings, and Appendix A fully discloses the exact system prompts used for training and inference. Our code and trained weights are also publicly released at `https://github.com/cvlab-stonybrook/omnigf`.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: Our code, including training and evaluation scripts with instructions for reproducing the results, and the trained model weights are publicly released at `https://github.com/cvlab-stonybrook/omnigf`. All benchmark datasets used in our experiments (GazeFollow, VideoAttentionTarget, ChildPlay, GOO-Real, GazeHOI, VSGaze) are publicly available, and the additional annotation files (e.g., our refined semantic labels for GazeFollow) will be released in the same repository.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Section 4.3 fully specifies the experimental setup, including VLM backbone, input resolutions, batch sizes, and optimizer, and Appendix B lists the dataset-specific learning rates and schedules.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: Due to the substantial computational resources required to fine-tune large Vision-Language Models, our experiments were conducted using a single fixed random seed. Consequently, error bars and statistical significance tests are not reported.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Section 4.3 details the type of compute resources, memory, and provides the estimated execution time range across datasets.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We have reviewed the NeurIPS Code of Ethics and confirm that the research conforms to the NeurIPS Code of Ethics. We solely utilize established, publicly available datasets, mitigating direct privacy and human subjects concerns. Furthermore, we explicitly address potential negative societal consequences, including dual-use and bias, along with corresponding impact mitigation measures in Appendix K.

   Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Appendix K (Broader Impacts) explicitly discusses both the positive societal impacts and the potential negative impacts of our proposed framework.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: While our framework utilizes a pre-trained VLM (Qwen3-VL), we release task-specific LoRA adapters and prediction heads strictly constrained to a specific task domain (gaze following). This artifact does not introduce new open-ended generative capabilities or the high risks for misuse (e.g., deepfakes, unrestricted text generation) that typically require specialized release safeguards.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.

- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All utilized assets, including our VLM backbone (Qwen3-VL), the head encoder initialization (ResNet-18 trained on Gaze360), and the evaluation datasets (GazeFollow, VideoAttentionTarget, ChildPlay, GOO-Real, GazeHOI, VSGaze), are properly credited via formal citations. We strictly respect their respective licenses: our released code inherits the GPL-3.0 license of the MTGS codebase it builds on, and our released weights are restricted to non-commercial use in accordance with the CC BY-NC 4.0 license of the Gaze360 head encoder.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The OmniGF codebase and trained weights are released at `https://github.com/cvlab-stonybrook/omnigf` with documentation covering installation, data preparation, training and evaluation commands, and license information. The refined semantic annotations for GazeFollow will be released in the same repository with documentation of their format.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: This paper relies exclusively on established, publicly available datasets and does not conduct any new crowdsourcing data collection or primary research involving human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: Our research relies entirely on the use of existing, publicly available datasets. As we did not conduct any original data collection or experiments involving human participants, institutional review board approval and subject risk disclosures were not required.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: The core ideas, experimental design, and technical contributions of this research were developed entirely by the authors. LLMs were utilized only as supportive tools for implementing code and refining the manuscript draft under the authors' strict supervision and verification.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.