Gaze Target Estimation Anywhere with Concepts

Supplementary Material

8. Further Discussions & Social Impact

8.1. Toward End-to-end Gaze Target Estimation

The evolution of human gaze estimation shows a clear trend: a move away from complex auxiliary features like pose and depth towards streamlined, head box-only inputs [10, 14, 19, 23, 24, 27, 35, 39, 44, 82]. This simplification has spurred the development of end-to-end, OpenPose [7]-like, and DETR [8]-like bottom-up approaches that can detect all head box-gaze pairs within a scene [18, 65, 66, 68]. However, a critical limitation persists. These methods lack identity association; they can find the gaze of everyone but cannot identify the gaze of a specific person. This necessitates separate modules or post-processing to link a detected gaze to a particular individual. Thus, the cascaded detection error will still exist. Our work, GazeAnywhere, directly addresses this gap. We propose the first text-promptable pipeline that simultaneously resolves human identification and gaze target estimation, enabling targeted queries for a specific person's gaze.

8.2. Future Application

Joint attention, the capability of following another person's head turn and gaze direction, typically emerges in children with Autism Spectrum Disorder (ASD) years later than in typically developing children [38]. Previous research has demonstrated the strong potential of gaze target estimation models to capture these atypical joint attention behaviors, offering a promising avenue for the early screening and detection of ASD [13, 43]. The concept-based prompting flexibility of the GazeAnywhere model offers a significant evolution in this domain. In future clinical and home-based settings, this model could be deployed to continuously and non-invasively track a child's gaze behavior. Critically, GazeAnywhere and GazeAnywhere Agent can be used by pediatricians or even the patient's parents simply by describing the patient's appearance or location in the prompt (e.g., "the child in the blue shirt"), thereby omitting the complicated and labor-intensive process of manually drawing head bounding boxes for annotation. This simplified usage offers the benefit of longitudinal tracking outside of a clinical setting, enabling earlier intervention and more comprehensive developmental monitoring. In addition, user can query GazeAnywhere Agent to let MLLM post-process the target tracking video and provide high-level gaze behavior information like gaze shift.

9. GazeAnywhere Agent

In this section, we introduce the GazeAnywhere Agent, a visual agentic framework designed to process natural-language gaze estimation and post-analysis requests. Figure 6 illustrate the workflow of the agent. The system dynamically queries a MLLM to orchestrate specific tools. The initial version of the agent integrates two primary models as the tool: Whisper-large-v3 for audio-to-text conversion and our proposed GazeAnywhere model for PGE target prediction.

Given an input image or video and a user request via audio , the MLLM acts as a planner and controller. It first converts the user's audio to text, analyzes the scene context, devises a step-by-step plan, and subsequently invokes the GazeAnywhere model. After each action, the agent receives visual feedback by visualizing the gaze target within the scene. This feedback is stored in memory, enabling the agent to revise its plan and determine the next steps for analysis. This pipeline handles queries far more complex than simple noun phrases, facilitating a deeper understanding of human gaze behavior in video streams.

10. Dataset & Benchmark

10.1. Training Set

The Gaze-Co training set contains 119,525 samples in total. Each record includes the target head bounding box, normalized gaze point, an in/out-of-frame label, and a compact concept phrase (attribute, position, action, and pose). The training data are constructed from three published gaze datasets after applying the image-quality filters, MLLM-based concept generation, and human in-loop MLLM verification described in the main text. In terms of source datasets, 69.6% (83,148 samples) come from GazeFollow [54, 55], 19.6% (23,481 samples) from VideoAttentionTarget [15], and 10.8% (12,896 samples) from ChildPlay [61] (see Fig. 7a).

For the apparent subject category, 51.0% (60,983 samples) are labeled as man, 32.2% (38,508) as woman, 8.2% (9,773) as boy, 6.1% (7,337) as girl, 1.6% (1,916) as child (unspecified gender), and 0.8% (1,008) as infant (unspecified gender) (Fig. 7b). These labels reflect perceived visual categories rather than verified identity attributes. Regarding gaze location, 13.9% (16,671 samples) of annotations are out-of-frame, while 86.1% (102,854) fall within the image (Fig. 7c).

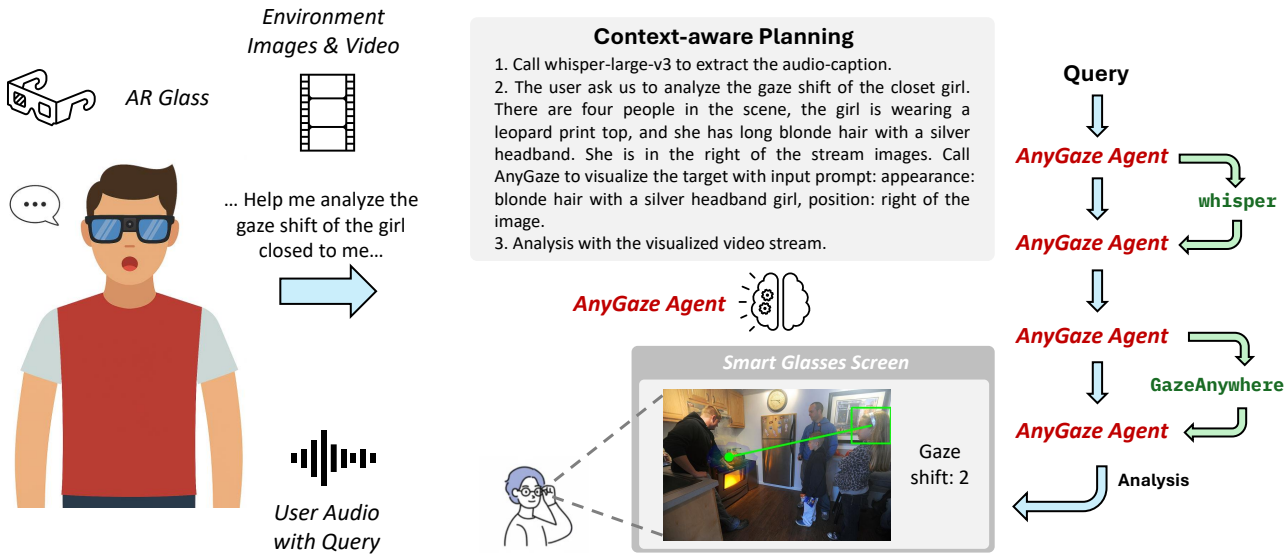
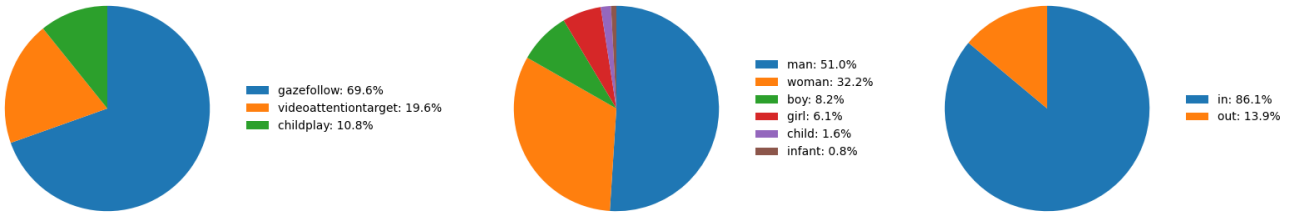


Figure 6. Step-by-step explanation of how GazeAnywhere Agent works.



(a) Source dataset distribution (b) Apparent subject category distribution (c) In-frame vs out-of-frame gaze distribution

Figure 7. Training-set statistics of the Gaze-Co dataset: (a) proportion of each source dataset, (b) distribution of apparent subject categories, and (c) proportion of in-frame vs out-of-frame gaze annotations.

10.2. Concept-based In-domain Test Set

We derive three concept-augmented test splits by converting the official test splits of GazeFollow, VAT, and ChildPlay into our unified PGE schema (image, head box, normalized gaze point, in/out-of-frame label, and concept phrase). After applying the same image-quality filters as in the training set, we obtain GazeFollow-Concept, VAT-Concept, and ChildPlay-Concept. To guarantee a high quality benchmark for both baselines and our model evaluation, all concept annotations are human verified instead of using MLLM.

GazeFollow-Concept. After filtering, GazeFollow-Concept contains 2,436 (image, head box) records. In terms of apparent subject category, 49.1% (1,197 samples) are labeled as man, 31.7% (772) as woman, 10.3% (250) as boy, 5.6% (136) as girl, 2.1% (50) as child (unspecified gender), and 1.3% (31) as infant (unspecified gender). All annotations in this split correspond to in-frame gaze targets (100%, 2,436 samples). In the dataset, each (image, head box) record is associated with multiple human gaze point annotations from the original GazeFollow dataset, which motivates the additional Avg L2 and Min L2 metrics used in the main text: Avg L2 is defined as the distance between the predicted gaze point and the mean of all human annotations, and Min L2 as the distance to the nearest human-annotated gaze point.

VAT-Concept. VAT-Concept contains 5,301 records. For apparent subject categories, 45.9% (2,435 samples) are labeled as man, 43.8% (2,324) as woman, 5.8% (310) as boy, 0.4% (20) as girl, 0.1% (5) as child (unspecified gender), and 3.9% (207) as infant (unspecified gender). Regarding gaze location, 35.5% (1,884 samples) of annotations are out-of-frame, while 64.5% (3,417) are in-frame.

ChildPlay-Concept. ChildPlay-Concept contains 1,238 records. In terms of apparent subject category, 8.5% (105 samples) are labeled as man, 32.9% (407) as woman, 36.8% (455) as boy, 12.0% (148) as girl, 4.4% (55) as child (unspec-

ified gender), and 5.5% (68) as infant (unspecified gender). For gaze location, 14.7% (182 samples) of annotations are out-of-frame, while 85.3% (1,056) are in-frame.
Across all splits, the apparent subject categories reflect perceived visual attributes rather than verified identity labels.

10.3. Concept-based Out-of-domain Test set

For out-of-domain evaluation, we utilize Child–Social Communication (Child-SC), a private dataset protected by IRB. It captures natural interactions between children and clinicians, where the clinician guides the child's attention across various targets using toys, thus eliciting frequent and structured gaze shifts. The dataset comprises 326 video clips from 40 children, sampled at 5 fps, yielding a total of 151,533 images. Due to privacy regulations, these images cannot be processed by cloud-based MLLM; consequently, all target-person concepts were manually annotated, strictly adhering to the style and protocols of our MLLM-generated concepts.

11. Baseline Details

11.1. Open-Vocabulary Detector (OVD)

As baselines, we use the OVD models to locate the target person described by a text prompt. This step supports our main task: to predict the point of view of the subject. Each OVD model takes an image and a prompt, matches text to visual regions in a shared vision–language space, and scores candidate boxes by text–image similarity. It outputs the highest-confidence bounding box for the prompted person, which we use as the subject-person localization. We also compared with the SOTA open-vocabulary human detection model RexSeek [31], which is a 3B foundation model in referring expression comprehension task.

GroundingDINO-B. GroundingDINO-B [42] is a Transformer-based detector featuring a dual-encoder single-decoder architecture that deeply fuses image and text features. It employs a language-guided query selection module to initialize object queries based on the input prompt. This mechanism produces a series of refined candidate boxes associated with prediction scores. From these outputs, we identify the target person by selecting the box with the highest confidence score for the referring phrase.

LLMDet-L. LLMDet-L [25] enhances open-vocabulary detection through multimodal co-training, where a large language model generates detailed captions to enrich feature alignment during training. At test time, with the LLM removed, the detector takes the image and prompt to generate multiple region candidates. It evaluates these regions by matching them against the text embedding, enabling us to filter the results and retrieve the top-ranked bounding box as the localized subject.

OWLv2-L. OWLv2-L [49] scales up the OWL-ViT architecture using a massive self-training strategy on over one billion weakly supervised examples. It utilizes a Vision Transformer backbone to directly predict bounding boxes and text-alignment scores from image tokens. When queried with the target person's description, the model outputs a collection of detected objects with their semantic similarity scores, from which we select the best-matching candidate to localize the person.

11.2. Gaze Model

Following the localization step, we evaluate several gaze-following models to predict the target person's point of regard. These models accept the full scene image and the localized person region as input. They output a 2D gaze heatmap (probability distribution), and we extract the coordinates of the peak value from the heatmap to represent the final predicted gaze location.

ViTGaze ViTGaze [60] is a single-modality gaze-following model that predicts a person's gaze target using RGB information only. Given the full image and the target person's head bounding box, it employs a pre-trained ViT to extract human–scene interaction cues directly from self-attention maps, eliminating the need for extra modalities. The model outputs a 2D gaze heatmap along with an in/out-of-frame score for evaluation.

Sharingan Sharingan [63] introduces a transformer-based architecture designed to capture global gaze interactions. It represents the target person via a Person Gaze Token, constructed by fusing head-crop features with normalized head-box coordinates. This token is processed with scene tokens by a ViT encoder to model human–scene dependencies. The model outputs a 2D gaze heatmap representing the spatial probability of the gaze target and an in/out-of-frame score.

Gaze-LLE Gaze-LLE [58] is a streamlined estimator built on a frozen, large-scale DINOv2 encoder, departing from traditional multi-branch head/scene architectures. Given the full image and the target person's head bounding box, it encodes the head location as a positional prompt injected into the scene features, using a lightweight transformer decoder to model head–scene relations. The model predicts a 2D gaze heatmap along with an in-/out-of-frame score.

12. Experimental Protocol

12.1. AR Device for GazeAnywhere Agent

We use DigiLens ARGO in the experiment to capture video data in real-world settings (Fig. 8). Its 48 MP camera records high-resolution video with autofocus, optical and electronic

Figure 8. DigiLens ARGO AR glasses used for video and audio capture and on-device feedback in the GazeAnywhere Agent.

stabilization, 4×4 pixel binning, and strong low-light support. For audio, a five-microphone beamforming array is designed to pick up the wearer's voice in noisy environments and provides spatial recordings suitable for analysis.

12.2. Implementation Details of GazeAnywhere-DINOv3-L

The deployed version of GazeAnywhere-DINOv3-L consists of a detector transformer with 3 layers and a dimension of $D = 256$. Both the visual and text prompts are trained jointly. For visual prompting, we apply diverse augmentation techniques during training, including head/body bounding box jittering, color jittering, random resizing and cropping, horizontal flipping, rotation, and masking of scene patches. For text prompting, as the subject position text information is fixed, we limit visual augmentation to random scene patch masking and apply text augmentation with reordering appearance, location, pose, and action attributes. During training, the input resolution is 512×512.

13. More Results

13.1. Impact of Frozen Encoder.

A key design choice for GazeAnywhere is to keep the image and text encoders frozen. We validate this approach in Table 7, which compares the default frozen model against one where the DINOv3 image encoder or the text encoder are fine-tuned. Unfreezing image or text encoders leads to a clear drop in performance. This demonstrates that DINOv3's pre-trained features are highly robust and generalizable for the PGE task, and that fine-tuning may lead to overfitting or harmful feature drift.

13.2. Impact of Detector Dimension.

We study the impact of the Detector transformer's layer dimension D in Table 8. The results indicate that performance plateaus at $D = 128$. We observed no significant performance gain from increasing D further, and thus selected

Visual	Text	Trainable Param	GazeFollow-Concept			VAT-Concept		
			AUC ↑	Avg L2 ↓	Min L2 ↓	AUC ↑	L2 ↓	AP ↑
✗	✗	870.2M	0.931	0.150	0.095	0.886	0.212	0.823
✓	✗	332.1M	0.943	0.119	0.067	0.886	0.206	0.813
✗	✓	541.7M	0.952	0.130	0.078	0.874	0.201	0.825
✓	✓	3.6M	**0.958**	**0.099**	**0.050**	**0.928**	**0.123**	**0.879**

Table 7. Comparison of the encoder frozen strategies.

D of $\psi(\cdot)$	GazeFollow-Concept			VAT-Concept		
	AUC ↑	Avg L2 ↓	Min L2 ↓	AUC ↑	L2 ↓	AP ↑
64	0.953	0.115	0.062	0.915	0.144	0.871
128	0.960	0.100	0.050	0.928	0.116	0.875
256	0.958	0.099	0.050	0.928	0.123	0.879
512	0.959	0.104	0.054	0.917	0.122	0.875

Table 8. Ablation experiment on the selection of the Detector Transformer dimension.

Layer Num of $\psi(\cdot)$	GazeFollow-Concept			VAT-Concept		
	AUC ↑	Avg L2 ↓	Min L2 ↓	AUC ↑	L2 ↓	AP ↑
1	0.949	0.126	0.076	0.882	0.179	0.846
2	0.957	0.104	0.054	0.920	0.123	0.871
3	0.958	0.099	0.050	0.928	0.123	0.879
4	0.957	0.100	0.051	0.928	0.120	0.858
5	0.959	0.095	0.047	0.929	0.121	0.888

Table 9. Ablation experiment on the selection of the transformer layer number in the Detector.

$D = 256$ as it provides the best trade-off between accuracy and computational cost.

13.3. Ablation on Detector's Transformer Layer Number

We conduct another ablation study to explore the layer number of transformer blocks in detector transformers. Results are shown in Table 9. After increasing the layer number to 3, the model shows stable performance.

14. Qualitative Analysis

In Figure 9, we qualitatively compare GazeAnywhere with the current state-of-the-art model, Gaze-LLE. Although Gaze-LLE performs well in sparse scenes with only one or two individuals, its performance degrades noticeably as crowd density increases. As shown in Figure 9, the upstream OVD module becomes unreliable in these complex settings and typically fails in two ways. First, it may localize the wrong person, causing Gaze-LLE to estimate gaze for an incorrect target. Second, it may produce an overly large bounding box that covers multiple people; even if the true target is included, Gaze-LLE cannot reliably disambiguate whom to condition on. These examples expose a key limitation of two-stage gaze estimation pipelines in real-world social scenes.

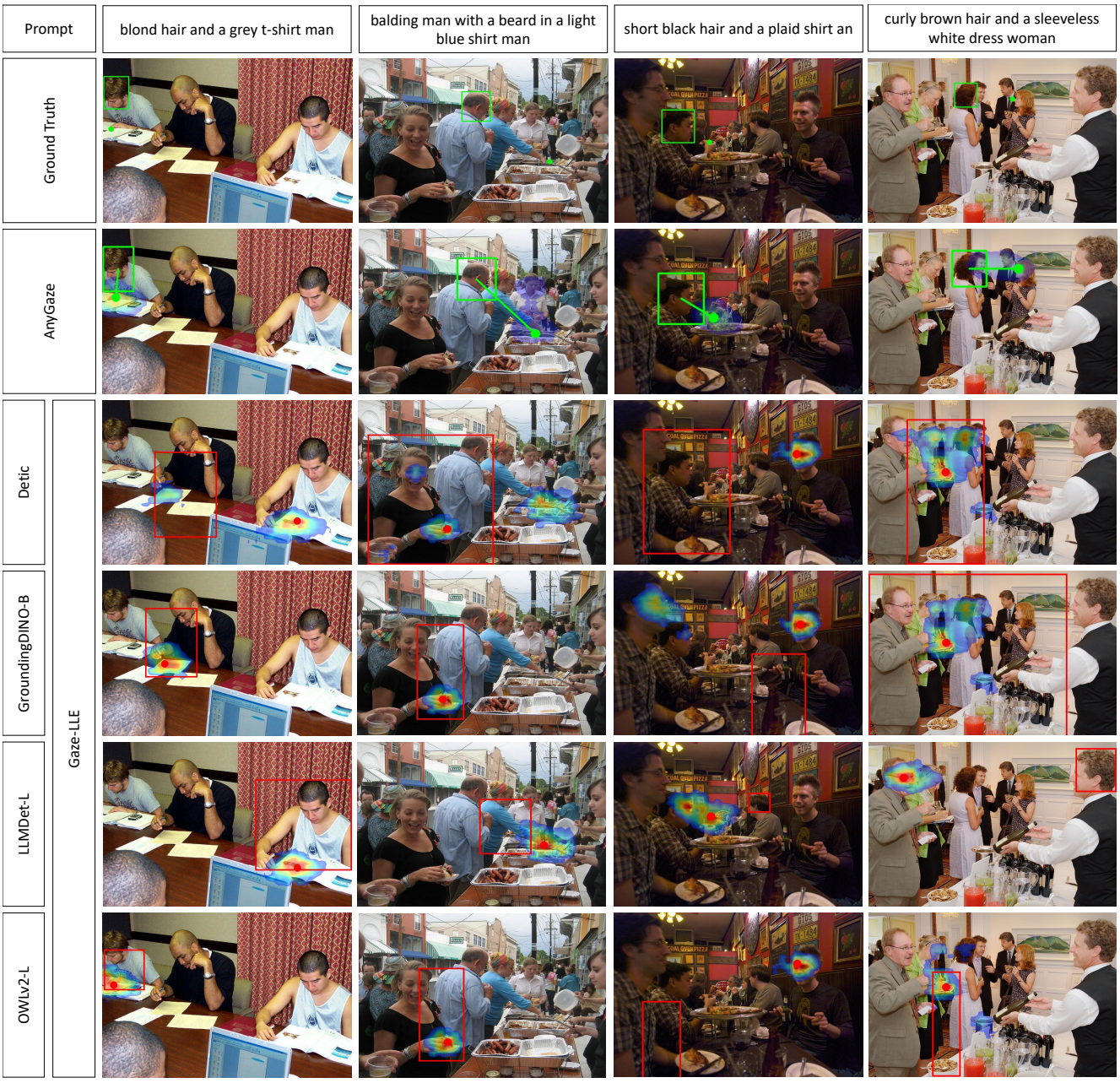


Figure 9. Qualitative comparison of gaze-target localization conditioned on appearance prompts. Each column corresponds to a different sample, and each row shows predictions from a different method. Our method produces sharper and more accurate heatmaps around the true gaze targets.

15. Related Prompts

For reproducibility, we include the exact natural-language prompts used to query the MLLM in our pipeline. These prompts support three major components: the concept-generation data engine, the MLLM-only gaze prediction baseline, and the GazeAnywhere Agent for video-based social gaze analysis. Unless otherwise noted, the prompts are shown verbatim as used in our batch API calls.

15.1. Data Engine

This section summarizes the prompts used by the data engine to construct concept-level annotations for each subject person. The attribute prompt (Fig. 10) instructs the MLLM to produce a compact description of appearance, position, action, pose, and people count for the person marked by the

green head box.

The concept verification prompt (Fig. 11) then asks the MLLM to check, field by field, whether a candidate concept matches the image and to return JSON flags for attribute, position, action, pose, and an overall pass/fail decision. Together with spot-checks from human annotators, these prompts implement the MLLM component of our human-in-the-loop data engine.

15.2. MLLM Baseline

Here we provide the prompt used for the MLLM-only gaze target prediction baselines, Gemini-2.5-flash (Fig. 12) and Qwen3-VL-8b (Fig. 13).

Given an image and a textual concept description, the model is asked to predict an gaze in/out-frame flag and a normalized 2D gaze target point, and to return the answer in a strict JSON format.

15.3. GazeAnywhere Agent

This section lists the prompts used to compare gaze-target analysis with an MLLM alone versus an MLLM assisted by the GazeAnywhere Agent on smart-glasses recordings. The raw-video prompt (Fig. 14) presents the model with the original AR recording and asks it to infer social gaze behavior directly from the unannotated video.

The GazeAnywhere-agent prompt (Fig. 15) uses the same video but with GazeAnywhere overlays (subject head box, gaze point, and out-of-frame indications), and instructs the model to count gaze shifts to social partners and overall gaze shifts.

16. Notations

We present the description all the notations in our paper in the last two pages.

Concept Generation Prompt

```
TASK
Return a description for the person with a green bounding box in head:
The description is a natural, concise attribute phrase (<30 words in total).

STYLE & CONTENT
- all lowercase
- avoid generic words:  person, people, adult; avoid starting with a/an/the
- prefer stable visible attributes, in this order:  hair style and color / hat >
glasses / beard > top garment color and pattern > pant / dress garment color and
pattern.
- the LAST word of attributes MUST be one of:  man, woman, boy, girl, infant, child
- prefer short and clear location description, such as bottom left corner.
- action:  describe ongoing interaction or movement; make it specific by adding
target/object/direction when visible.  keep “what is being done” here.  if unclear,
write "none".
- pose:  describe static body configuration and facing direction; keep “how the body
is” here (orientation, posture, limb arrangement).  if unclear, write "none".
- keep action and pose distinct and non-overlapping.
- also provide an approximate count of people visible in the scene; report a single
integer when feasible; if indeterminate, write "none".

You output format is:
<attribute> attributes of the human </attribute>
<position> position of the human in the camera </position>
<action> action of the human </action>
<pose> pose of the human </pose>
<count> estimated number of people in the scene </count>
```

Figure 10. **Concept generation prompt** used for generating concept phrases for the target person.

Concept Verification Prompt

```
TASK
You see an image with one green bounding box on a human head and a candidate
description from Prompt A, with:
<attribute>, <position>, <action>, <pose>, <count>.
Check whether the first four fields match the target human and the scene.  Ignore
<count>.


CHECKING RULES

attribute
must describe the same person in the green box.
hair / hat / glasses / beard / clothes must match.
last word must be one of:  man, woman, boy, girl, infant, child.
label as correct only if all above are satisfied.


position
must match the boxed human location (e.g., top left, bottom center, center right).
label as correct only if consistent with boxed human position.


action
must be a visible ongoing movement or interaction of this person.
if not clearly visible, the correct value should be "none".
label as correct only if supported by the image.


pose
must describe static body configuration and facing direction (orientation, posture,
limb arrangement).
must be distinct from action; if unclear, should be "none".
label as correct only if supported by the image and distinct from action.


OVERALL
overall is "pass" only if all four checks are "correct".
otherwise overall is "fail".


OUTPUT FORMAT
Output only a single JSON object with exactly these keys and values:

"attribute_check":  "correct" or "incorrect"
"position_check":  "correct" or "incorrect"
"action_check":  "correct" or "incorrect"
"pose_check":  "correct" or "incorrect"
"overall":  "pass" or "fail"
```

Figure 11. **Concept verification prompt** used to check attribute, position, action, and pose consistency for each subject.

Gaze Target Prediction Prompt of Gemini 2.5 Flash

```
You are given an image, where the top-left corner is (0, 0) and the bottom-right
corner is (1, 1).

Coordinates are normalized by the image width and height.
You are also given a description of the subject person in the image:

Attribute: {attribute}
Location: {position}
Action: {action}
Pose: {pose}
Based on this description and the image, perform the following tasks:

1. In-frame gaze flag
Indicate whether the subject person is looking at a target inside the image frame.

Output 1 if the gaze target lies within the image frame.
Output 0 if the subject person is looking outside the image frame.

2. Gaze target point
Predict the gaze target of the subject person as a point (x, y) ∈ [0, 1], with exactly
three decimal places for both x and y.

The values must be normalized by the image width and height.

Output format
Return only a valid JSON object, with no extra text, in the following format:
{
"in_frame_gaze": 0,
"gaze_target": {
"x": 0.XXX,
"y": 0.XXX
}
}
in_frame_gaze must be either 0 or 1.
x and y must be numbers in [0,1] with three decimal places.
```

Figure 12. **Gaze target prediction prompt** used for in-frame flagging and point estimation on Gemini-2.5 baseline

Gaze Target Prediction Prompt of Qwen3-VL.

```
You are given an image, where the top-left corner is (0.000 , 0.000), the bottom-right
corner is (1.000, 1.000).  The pixel point in the image is normalized to 0.000 to
1.000.  All values are rounded by 3.

Here is the description of the subject person Question

Based on the description of a subject person in the image, perform the following task:

(1) Indicate whether the subject person is looking at a target inside the image frame.

Output 1 if the gaze target lies within the image frame.  Output 0 if the subject
person is looking outside the image frame.

Provide the inside prediction between the <inside> and </inside> tags.

(2) Predict the gaze target of the subject person as a point (x, y) x and y are in
[0.000, 1.000].  If looking outside, randomly give values.

Provide the x of point between the <x> and </x> tags, y of point between the <y> and
</y> tags.
```

Figure 13. **Gaze target prediction prompt** used for in-frame flagging and point estimation on Qwen3-VL-8B.

Gaze Shift Analysis Prompt on Single MLLM Solution

```
This is a short video for human gaze target understanding.  Can you give me an
analysis of these tasks"?

The subject child is:  short black hair white dress girl

1.  "Gaze shift to social partner":  The number of gaze shifts to the nearby person
happened.

2.  "Total gaze shift count":  The number of gaze shifts happened.  (change the gaze
target to another object or out-of-frame)

Hint:  Gaze shifting is the coordinated movement of the eyes and head to look at a
new target.  Per gaze shift means changing the gaze target from one object/human to
another object/human

You should analyze the video and provide the answers to the above tasks.
```

Figure 14. **Gaze shift analysis prompt** used for counting gaze shifts and eye contact events on Single MLLM.

Gaze Shift Analysis Prompt on GazeAnywhere Agent

```
This is a short video for human gaze target understanding.  The green bounding box is
the detected subject child.  If the bounding box's color becomes blue, it indicates
the subject is looking out of the frame.  The green point is the child's gaze target,
and we overlap it with the raw video.  Can you give me an analysis of these tasks"?

The subject child is:  short black hair white dress girl

1.  "Gaze shift to social partner":  The number of gaze shifts to the nearby person
happened.

2.  "Total gaze shift count":  The number of gaze shifts happened.  (change the gaze
target to another object or out-of-frame)

Hint:  Gaze shifting is the coordinated movement of the eyes and head to look at a new
target.  Our green point in the frame can indicate the target location.  So you should
infer if the target is changed.  Per gaze shift means changing the gaze target from
one object/human to another object/human

You should analyze the video and provide the answers to the above tasks.
```

Figure 15. **Gaze shift analysis prompt** used for counting gaze shifts and eye contact events on GazeAnywhere agent.

Data and Indices	
H	Height of input image
W	Width of input image
$I \in \mathbb{R}^{3\times H\times W}$	Input RGB image
P	Prompt
T	Text
H_{out}	Height of output image
W_{out}	Width of output image
$\hat{H} \in \mathbb{R}^{H_{out}\times W_{out}}$	Gaze heatmap
Embeddings and Image Encodings	
$\phi_V(\cdot)$	Image encoder
N_V	Number of patch tokens
D_V	Visual embedding dimension
$[CLS]$	Classification token
$c \in \mathbb{R}^{D_v}$	$[CLS]$ token embedding
$s_i \in \mathbb{R}^{D_V}$	Visual output embedding token
Embeddings and Text Encodings	
$\phi_T(\cdot)$	Text encoder
$[EOS]$	End of sentence token
T_E	Initial text embeddings
L_T	Fixed context length
D_T	Text embedding dimension
t_{eos}	End of sentence token
t_{pad}	Padding token
$t_i \in \mathbb{R}^{D_T}$	Text embedding token
Projection Layers	
$W_V \in \mathbb{R}^{D_V\times D}$	Trainable visual linear projection layer
$W_T \in \mathbb{R}^{D_T\times D}$	Trainable test linear projection layer
D	Projected dimension
$Z_V \in \mathbb{R}^{(1+N_V)\times D}$	Projected visual tokens
$Z_T \in \mathbb{R}^{L_T\times D}$	Projected text tokens
Detector Transformer	
$\psi(\cdot)$	Detector transformer
$t_h \in \mathbb{R}^D$	Head token
$t_p \in \mathbb{R}^D$	Target presence token
c'	Projected global visual tokens
t'_{eos}	Projected global text tokens
E_{head}	Learnable head embeddings
$E_{presence}$	Learnable presence embeddings
$s' \in \mathbb{R}^{N_V\times D}$	Projected visual patch tokens from Z_V (excluding c')
$t' \in \mathbb{R}^{(L_T-2)\times D}$	Projected text patch tokens from Z_T (excluding t'_{eos} and padding)
$F \in \mathbb{R}^{(N_T+N_V+2)\times D}$	Full input sequence F
$\psi(F) \in \mathbb{R}^{(N_T+N_V+2)\times D}$	Output refined sequence of detector transformer
Decoder	
$\hat{s} \in \mathbb{N}_\mathbb{V} \times \mathbb{D}$	Refined visual patch tokens
$\hat{t_h} \in \mathbb{R}^D$	Refined head tokens
x	normalized center x coordinate of head tracker
y	normalized center y coordinate of head tracker

w	normalized width of head tracker
h	normalized height of head tracker
$\hat{t_p} \in \mathbb{R}^D$	Refined predict tokens
Learning Objective	
$\mathcal{L}_{total}$	Total loss
$\mathcal{L}_{gaze}$	Gaze heatmap BCE loss
σ	Standard deviation of 2D Gaussian
$\hat{Y}$	Predicted heatmap
N	Total number of pixels
p	Single pixel on the heatmap
y_p	Ground-truth of p
$\hat{y_p}$	Predicted values of p
$\mathcal{L}_{presence}$	Target presence focal loss
$Y_{presence} \in 0, 1$	Ground truth target presence
$\hat{Y_{presence}} \in [0, 1]$	Predicted target presence
$\mathcal{L}_{focal}$	Focal loss
$\mathcal{L}_{head}$	Head box loss
$\mathcal{L}_1$	Mean absolute error
$\mathcal{L}_{IoU}$	GIoU loss
b	Ground truth head box
$\hat{b}$	Predicted truth head box
λ_{l_1}	head object detection hyperparameter
λ_{IoU}	head object detection hyperparameter
Data Engine	
x_{min}	x coordinate of top-left corner of the head box
y_{min}	y coordinate of top-left corner of the head box
x_{max}	x coordinate of bottom-right corner of the head box
y_{max}	y coordinate of bottom-right corner of the head box
g_x	x coordinate of ground truth gaze point
g_y	y coordinate of ground truth gaze point