PAGE: TOWARDS PRACTICAL HUMAN-LEVEL GAZE TARGET ESTIMATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Gaze target estimation, the task of predicting where a person is looking in a scene, is crucial to understanding human attention and intent. It is a challenging task that combines high-level understanding of global scene semantics and precise spatial reasoning using human appearance (e.g. pose, eye orientation). As a result, human-level performance remains elusive for existing models, limiting their practical application. To this end, we propose PaGE (**Pra**ctical **G**aze **E**stimator), a gaze estimation model that explicitly models the complex interaction between scene and head features through repeated cross attention. Using a PaGE model with a large ViT-H+ backbone as the teacher, we further distill student models with lighter backbones on a much larger and more diverse unlabeled dataset. The architectural improvements and novel training recipe allow PaGE to achieve state-of-the-art performance on several gaze estimation tasks, outperforming humans in 7 out of 9 metrics while reducing the human-AI gap by at least 60% in the remaining 2. The distilled student models retain most of the teacher's performance while being lightweight enough for practical deployment on robots and consumer devices. The code is available in the supplementary materials.

1 INTRODUCTION

Gaze is one of the most important non-verbal social cues. It provides valuable insight into a person's attention and intent, as well as the dynamics of social interactions. It is a key component of socially-aware interactive systems like MLLM agents and robots. Humans perform accurate *gaze following* (i.e., identifying the gaze target of another person in a scene) naturally, yet it is challenging to replicate this capability with vision models. This can be attributed to the inherent complexity of gaze following—it requires a combination of scene understanding and accurate spatial reasoning using human appearance cues (e.g., pose, eye orientation). Therefore, existing models perform substantially worse than humans, limiting their practical application in fields like HCI and robotics.

In this work, we propose PaGE (Figure 1), **the first gaze estimation model with human-level performance**. On GazeFollow, VideoAttentionTarget and ChildPlay, PaGE outperforms humans on 7 out of 9 metrics while closing the current human-AI gap by at least 60% for the remaining two. Distilled versions of PaGE retain SOTA performance while being lightweight enough for **real-time gaze following on robots and many consumer devices**.

The strong results come from a combination of (1) a **novel model architecture** designed to explicitly model feature interaction between the scene and head branches, and (2) **improvements to the training recipe**, an area underexplored in previous work. Specifically, we propose the Scene-head Interaction Module (SIM), a novel gaze decoder component that uses cross attention between the scene and head branches to explicitly model inter-branch feature interaction in a ViT-native manner. This affords PaGE the spatial reasoning capability needed to pinpoint the gaze target. For training, we adopt a new two-stage approach. We first train the decoder only, with the backbone frozen. We then finetune the entire model, backbone included, to further adapt the model to gaze prediction tasks. To build strong lightweight models, we further propose a token-level feature distillation procedure that trains student models using a PaGE ViT-H+ teacher. To ensure that the student learns a generalizable feature representation for gaze estimation, we leverage large-scale image data without gaze annotation for distillation as a remedy to the scarcity of labeled data. This results in a line of SotA lightweight PaGE models with $\sim 10\%$ of the teacher's FLOPs that are practical to deploy.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Apart from proposing PaGE, we also dissect how each component of our architecture and training recipe impacts performance, providing insight for designing future models. We further demonstrate PaGE's versatility by adapting it to fine-grained perception of gaze on screens, as well as discussing its potential as a universal visual attention estimator beyond human gaze and realistic images.

2 Related Work

Gaze following is a complex and demanding task requiring a combination of global scene understanding and fine-grained human appearance perception capabilities. One rather intuitive approach is to dedicate a branch to each capability before fusing the features for a final prediction. In addition to the main scene branch, such *multi-branch* models may also include branches for a head crop (Recasens et al., 2015; Chong et al., 2020; Miao et al., 2023), depth (Fang et al., 2021; Bao et al., 2022; Jin et al., 2022; Gupta et al., 2022), and pose (Bao et al., 2022; Jin et al., 2022; Gupta et al., 2022).

Another approach emerged as general-purpose vision backbones (e.g., CLIP, DINO) demonstrated state-of-the-art performance in dense vision tasks like depth estimation and segmentation (Siméoni et al., 2025). In this line of work (Tafasca et al., 2024; Song et al., 2024; Ryan et al., 2025), the combination of capabilities is packed into a single large-scale pre-trained backbone. The focus shifts to designing effective gaze decoding strategies. Tafasca et al. (2024) used a DPT decoder conditioned by head features. Song et al. (2024)'s decoder leverages both patch features and attention maps produced by the ViT encoder. Gaze-LLE (Ryan et al., 2025) uses a ViT stack to decode the heatmap from DINOv2 patch features, injecting head position by adding a learnable head prompt to relevant patch tokens. AnyGaze (Cao et al., 2026b) explores text prompts as an alternative to head bounding boxes by leveraging an additional pre-trained text encoder. This line of work has led to more flexible, better-performing models than previous multi-branch ones, but is still far from reaching human performance. However, in contrast to recent architectural progress, few have attempted to improve the training recipe of gaze estimation models built upon general-purpose backbones.

Recent progress in gaze target estimation has largely been *model-driven*. Datasets have remained more or less the same, with most work using GazeFollow (Recasens et al., 2015) and VideoAttentionTarget (VAT) (Chong et al., 2020). Some recent work also uses ChildPlay (Tafasca et al., 2023) and GOO (Tomas et al., 2021). While other datasets have been proposed (Hu et al., 2023), they have yet to gain traction. An obstacle to *data-driven* progress is the tension between accurate annotation and diversity. Datasets like GazeFollow cover diverse scenes sourced from the web but rely on somewhat unreliable human annotations. Meanwhile, datasets with accurate ground truth must be collected in controlled environments under strict procedures (Hu et al., 2023; Tomas et al., 2021).

In this work, we combine the strengths of multi-branch models and the DINOv3 (Siméoni et al., 2025) backbone to create PaGE. Our core architectural contribution is a ViT-native solution for explicit feature interaction between the scene and head branches. We also innovate in how we train gaze models built upon general-purpose backbones, an underexplored area, by introducing SFT and distillation. We further demonstrate that PaGE is limited by data and has untapped model capacity that can be readily adapted to challenging tasks if trained on proper data.

3 PaGE

3.1 Problem Definition

2D third-person gaze target estimation can be formally defined as follows: given an RGB image $x_{img} \in \mathbb{R}^{3\times H_{in}\times W_{in}}$, which must contain at least one person, and a head bounding box $x_{bbox} \in \mathbb{R}^4$, the model predicts a heatmap $H \in \mathbb{R}^{H_{out}\times W_{out}}$ where $H_{ij} \in [0, 1]$. The model also predicts $P_{in} \in [0, 1]$, the probability that the person's gaze target is inside the image. An alternative detection-based formulation (Tu et al., 2022; 2023; Tonini et al., 2023) is discussed in Appendix M.

3.2 Building PaGE

Figure 1 provides an overview of the PaGE architecture. In this section, we walk through the design choices we made for PaGE. We start from Gaze-LLE (Ryan et al., 2025), the previous SOTA model, and introduce the changes we made to the backbone, training recipe, and decoder architecture one

by one. Gaze-LLE is chosen as the starting point because of its effectiveness and simplicity among prior work. The performance after each change is shown in Table 1.

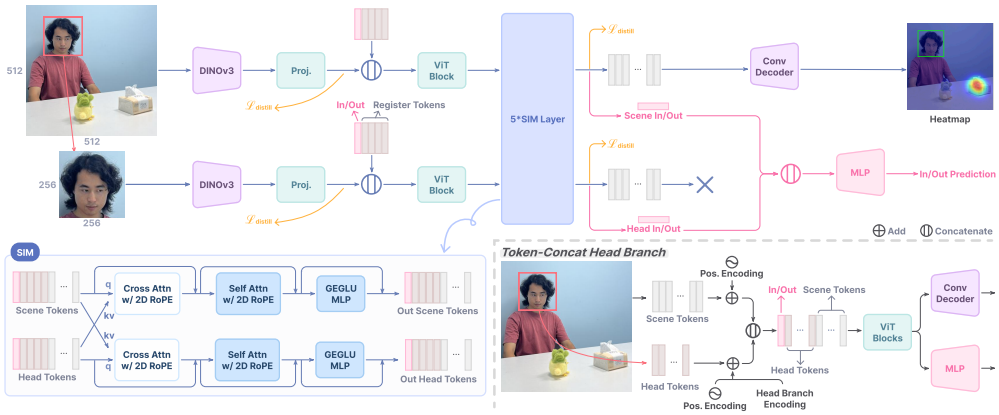

Figure 1: Architecture of PaGE. Bottom left: our SIM layer. Bottom right: The *token-concat* head branch baseline. We also mark where the $\mathcal{L}_{distill}$ loss is applied during distillation.

	GazeFollow			VideoAttentionTarget			ChildPlay		
	AUC↑	Avg L2↓	Min L2↓	AUC↑	L2↓	APin/out↑	AUC↑	L2↓	APin/out↑
Gaze-LLE ViT-B†	0.9543	0.1104	0.0492	0.9324	0.1074	0.8970	0.9478	0.1051	0.9920
Simplify Training Recipe	0.9552	0.1074	0.0482	0.9370	0.1078	0.8926	0.9499	0.1103	0.9907
DINOv2 → DINOv3	0.9554	0.1044	0.0450	0.9383	0.1118	0.9053	0.9496	0.1062	0.9927
Token-concat Head Branch	0.9581	0.1003	0.0439	0.9462	0.1000	0.9138	0.9585	0.0955	0.9942
3 Layers → 5 Layers	0.9583	0.0985	0.0419	0.9516	0.0928	0.9150	0.9604	0.0907	0.9949
DINO Feature Dropout	0.9591	0.0979	0.0412	0.9502	0.0912	0.9235	0.9596	0.0911	0.9945
SIM Head Branch	0.9596	0.0961	0.0392	0.9543	0.0911	0.9304	0.9618	0.0905	0.9949
Register Tokens	0.9595	0.0968	0.0403	0.9544	0.0878	0.9242	0.9608	0.0868	0.9931
GEGLU MLP	0.9596	0.0957	0.0399	0.9546	0.0858	0.9277	0.9625	0.0873	0.9944
2D RoPE	0.9599	0.0939	0.0380	0.9557	0.0886	0.9421	0.9652	0.0883	0.9961
Finetune (PaGE ViT-B)	0.9625	0.0892	0.0345	0.9618	0.0844	0.9491	0.9690	0.0810	0.9971
Gaze-LLE ViT-L†	0.9569	0.1058	0.0455	0.9371	0.1035	0.9014	0.9505	0.1010	0.9926
PaGE ViT-L	0.9648	0.0823	0.0306	0.9690	0.0736	0.9460	0.9728	0.0724	0.9957
Human‡	0.924	0.096	0.040	0.921	0.051	0.925	0.911	0.048	0.993
PaGE ViT-H+	0.9659	0.0804	0.0288	0.9719	0.0643	0.9509	0.9746	0.0687	0.9954

Table 1: The road to PaGE. We track the impact of each modification we made. †Reevaluated with official checkpoints (the "inout" version for GazeFollow and VAT, the "inout_childplay" version for ChildPlay) and code. ‡Human baseline reported by dataset authors (Recasens et al., 2015; Chong et al., 2020; Tafasca et al., 2023), only three decimal digits are available.

3.2.1 Simplified Training Recipe

Prior work on gaze estimation trains the model on GazeFollow first before finetuning it on VAT and ChildPlay (Tafasca et al., 2024; Ryan et al., 2025). This approach is unnecessarily complex and difficult to integrate with our finetuning (Section 3.2.5) and distillation (Section 3.3) steps. We take a simplified approach instead, training the model on a combined dataset consisting of images from GazeFollow, VAT, and ChildPlay. Specifically, we use the full train split of GazeFollow, sample 1 out of every 3 VAT frames, and 1 out of every 6 ChildPlay frames. We sample VAT and ChildPlay at a reduced rate because adjacent video frames are highly similar (see Appendix B for ablation). An added benefit is that this approach produces a single set of model weights instead of one checkpoint for each dataset, simplifying evaluation and downstream use. We use the standard multitask loss for gaze target estimation: $\mathcal{L} = \mathcal{L}_{heatmap} + \lambda\mathcal{L}_{in/out}$, where $\mathcal{L}_{heatmap}$ and $\mathcal{L}_{in/out}$ denote BCE Loss. We empirically choose $\lambda = 0.01$ for all experiments. The rest of the training recipe is detailed in Appendix A.1. We use the checkpoint from the final epoch for evaluation, finetuning and distillation. As shown in Table 1, the new, simpler training recipe does not impact the results too much. **All remaining experiments in Section 3.2 are done with this recipe, and we refer to this train set as the *labeled train set* from now on.**

3.2.2 DINOv3 BACKBONE

We replace DINOv2 (Oquab et al., 2024) with DINOv3 (Siméoni et al., 2025), a newer and stronger backbone. Since the patch size of DINOv3 is 16 instead of DINOv2's 14, we increase the input resolution from 448×448 to 512×512 to keep the number of patches the same. This simple drop-in replacement results in a modest improvement (Table 1). For more substantial gains, we would need to redesign the decoder architecture. **We use DINOv3 and 512×512 input from now on**. See Appendix C for experiments with alternative backbones (e.g., CLIP, TIPSv2, etc.).

3.2.3 WE DO NEED A HEAD BRANCH

Ryan et al. (2025) found that adding a head branch to Gaze-LLE does not bring significant improvement and hypothesized that DINOv2 extracts sufficient head features from the scene image alone. However, they only explored a flawed implementation where the scene and head feature maps extracted by DINO are directly concatenated along the *channel* dimension. This creates a **spatial misalignment**. For example, features from the top left of the head crop would be concatenated with features from the top left of the scene. While prior models using convnets (Chong et al., 2020; Miao et al., 2023) also took this suboptimal approach, the attention mechanism in ViTs allows for more flexibility regarding branch fusion. **In this section, we show that a well-designed ViT-native head branch, central to our architectural contribution, substantially improves gaze estimation**.

We first consider a head branch architecture that requires the least change (Figure 1, bottom right). Specifically, we extract patch tokens from a 256×256 head crop with DINOv3, which are then concatenated with scene tokens along the *length* dimension. We then add a learnable *head branch encoding* to all head tokens to distinguish them from scene tokens. Finally, we pass the tokens through a ViT stack identical to Gaze-LLE's. This *token-concat* head branch significantly improves all metrics (Table 1), suggesting that **a head branch is indeed needed in our quest for human-level gaze following**. We also find that the head branch helps with *scaling*. While Ryan et al. (2025) did not find any upside in stacking more than 3 decoder layers, increasing the number of decoder layers from 3 to 5 is effective for the token-concat model. **We use 5 decoder layers from now on.** We then apply channel dropout with $p = 0.1$ on the DINOv3 feature maps during training. Interestingly, this simple change leads to measurable performance gains, suggesting that DINOv3 could cause our larger, deeper decoder to overfit when used off-the-shelf. **Channel dropout on backbone features is applied to all models from this point onward**.

Despite its effectiveness, the token-concat approach still has room for improvement. It forces scene features and head features into the same feature space by concatenating the branches into the same token sequence. This motivated us to design the Scene-head Interaction Module (SIM, Figure 1, bottom left), which keeps scene and head features in separate branches and explicitly models inter-branch feature interaction with cross attention instead. After cross attention with the other branch, both branches are passed through a standard ViT block consisting of self attention and an FFN. A stack of SIMs could then model complex interactions between scene and head features. Finally, considering that we have two separate branches now, we append an In/Out token to both branches and concatenate them to create the input for the In/Out prediction head. The results in Table 1 show that decoupling the two branches is effective. Results on alternative SIM architectures are in Appendix E. **All decoders from now on use SIM layers instead of vanilla ViT blocks.**

3.2.4 MODERNIZING THE VIT DECODER

Gaze-LLE uses vanilla ViT blocks as its decoder. Although we replaced ViT blocks with SIM modules in Section 3.2.3, SIM itself is still made up of plain ViT components. In this section, we introduce common modern ViT components. We first add 4 register tokens to the scene and head branches. We then replace the standard GELU MLPs with GLU-style MLPs. We tried SwiGLU and GEGLU, and GEGLU performed better (Appendix F.2). **We keep these changes in SIM.**

Finally, we replace the 2D sinusoidal positional encoding with 2D RoPE, which is also used in DINOv3. For self attention, the standard axial 2D RoPE implementation proposed by Heo et al. (2025) is used. For cross attention, the scene branch uses standard RoPE, while the head branch calculates RoPE based on the scene branch coordinates of the head crop. See Appendix D for the exact formulation. An added benefit of 2D RoPE is that it **implicitly encodes head position**, as both branches operate in the same coordinate system. Therefore, we remove the *learnable head prompt*

introduced by Gaze-LLE. As shown in Table 1, 2D RoPE is especially effective on GazeFollow, as well as predicting whether the gaze target is in-frame ($AP_{in/out}$). **We have now arrived at the final architecture of PaGE, a strong and modern gaze target estimation model. It will be used in the rest of this paper.** See Appendix F for more extensive ablation studies of the architecture.

3.2.5 FINETUNING THE BACKBONE

Until now, we have kept the DINOv3 backbone frozen. This is because making the backbone trainable from the start actually degrades performance (see Section 4.2 for experiments). However, a short supervised finetuning (SFT) regime with the backbone unfrozen *after* the decoder is trained does bring concrete improvements. Specifically, we use the same labeled train set, and train the scene and head branch backbones separately to allow for branch-specific feature adaptation, although both backbones start from DINOv3 weights. The detailed finetuning procedures are in Appendix A.1. As shown in Table 1, SFT significantly improved performance on all metrics. **The finetuned ViT-H+ variant, in particular, achieves human-level performance, outperforming humans on 7 out of 9 metrics. It is now ready to be used as the teacher model for distillation.**

3.3 FEATURE DISTILLATION WITH UNLABELED IMAGES

In previous sections, we achieve strong performance when a heavy ViT-H+ (840M Params) backbone is used. However, the results of smaller variants (e.g., PaGE ViT-B), while still SOTA, lag behind those of PaGE ViT-H+ and humans. Therefore, in this section, we explore how to **transfer the superior capabilities of PaGE ViT-H+ to smaller variants through feature distillation.**

3.3.1 DATASET CURATION

Knowledge distillation works well when the dataset is large and diverse (Frank & Davis, 2025). Considering that labeled gaze datasets are limited in both of these regards, we turn to public image datasets without gaze annotation instead. Specifically, we combine MPII (Andriluka et al., 2014), a human pose dataset with high quality images of people, and OpenImages V7 (Krasin et al., 2017), a much larger dataset with object annotations. The resulting distillation set consists of all MPII training samples and all OpenImages V7 training samples containing objects labeled "*person*", "*man*", "*woman*", "*boy*", and "*girl*". We do not use MS COCO (Lin et al., 2014), a major source of GazeFollow test images, to avoid data leakage. We then follow Ryan et al. (2025) and use a YOLOv5 head detector to provide head bounding boxes for people inside each image. For each image, we keep a maximum of 3 heads to ensure scene diversity. The final distillation set consists of $1.17M$ heads from $608k$ images, significantly more than the $200k$ samples in our labeled train set. A simple experiment validating the quality of the unlabeled distillation set is in Appendix G.

3.3.2 DISTILLATION OBJECTIVE

We conduct token-level feature distillation to train lightweight student models that retain the PaGE ViT-H+ teacher's strong performance. The main objective is to align the student's final-layer (i.e., after the last SIM layer) token representation to the teacher's. Formally, for an input pair $\{x_{img}, x_{bbox}\}$, let $\mathbf{z}^s, \mathbf{z}^t \in \mathbb{R}^d$ denote the student and teacher tokens, respectively. We optimize a distillation loss that combines element-wise ℓ_1 loss with a cosine alignment term:

$$\mathcal{L}_{\text{distill}} = \left\|\mathbf{z}^s - \mathbf{z}^t\right\|_1 + \left(1 - \frac{\langle \mathbf{z}^s, \mathbf{z}^t \rangle}{\|\mathbf{z}^s\|_2 \, \|\mathbf{z}^t\|_2}\right).$$

The ℓ_1 term encourages absolute feature matching, while the cosine term encourages directional alignment. The loss is applied to both scene and head branch tokens. We then add an *auxiliary objective* that aligns the student's *backbone features* to the teacher's right after they are both projected to the decoder dimension (Figure 1). The idea is to prevent representation collapse in the student backbones. We use the same $\mathcal{L}_{distill}$ objective and add the auxiliary loss directly to the main loss. We train both the backbone (initialized with DINOv3 weights) and the PaGE decoder (random initialization). As in Section 3.2.5, the scene and head backbones are trained separately to allow for branch-specific feature adaptation. Since distillation only teaches the student feature representation, not downstream prediction, we add a short post-distillation finetuning regime on the labeled train set. The exact procedure and hyperparameters are detailed in Appendix A.2.

Model	GFLOPs	GazeFollow			VideoAttentionTarget			ChildPlay		
		AUC	Avg L2	Min L2	AUC	L2	AP	AUC	L2	AP
PaGE ViT-S	96.9	0.9605	0.0978	0.0411	0.9516	0.0954	0.9218	0.9641	0.0899	0.9961
PaGE ViT-S Distill		0.9643	0.0858	0.0327	0.9640	0.0739	0.9371	0.9701	0.0751	0.9966
PaGE ViT-S+	115.2	0.9611	0.0930	0.0370	0.9548	0.0919	0.9302	0.9652	0.0868	0.9963
PaGE ViT-S+ Distill		0.9647	0.0861	0.0331	0.9654	0.0740	0.9388	0.9704	0.0746	0.9971
PaGE ViT-B	283.1	0.9625	0.0892	0.0345	0.9618	0.0844	0.9491	0.9690	0.0810	0.9971
PaGE ViT-B Distill		0.9660	0.0814	0.0295	0.9688	0.0677	0.9450	0.9734	0.0697	0.9969
PaGE ViT-H+ (Teacher)	2373.6	0.9659	0.0804	0.0288	0.9719	0.0643	0.9509	0.9746	0.0687	0.9954

Table 2: Comparison between PaGE models trained from scratch and ones distilled from PaGE ViT-H+ teacher. Distilled models perform significantly better and retain most of the teacher's capability.

We report the performance of the distilled models in Table 2. The distilled models performed significantly better than identical models trained from scratch and retained most of the teacher's performance. This proves that feature distillation enables *efficient* human-level gaze target estimation with roughly 5%-10% of the teacher's compute burden. Detailed profiling is in Appendix H.

4 Further Experiments and Results

4.1 Full Results

Model	GazeFollow			VideoAttentionTarget			ChildPlay		
	AUC↑	Avg L2↓	Min L2↓	AUC↑	L2↓	$AP_{in/out}$↑	AUC↑	L2↓	$AP_{in/out}$↑
Recasens et al. (2015)	0.878	0.190	0.113	-	-	-	-	-	-
Chong et al. (2018)	0.896	0.187	0.112	0.833	0.171	0.712	-	-	-
Lian et al. (2019)	0.906	0.145	0.081	-	-	-	-	-	-
Chong et al. (2020)	0.921	0.137	0.077	0.860	0.134	0.853	-	-	-
Fang et al. (2021)	0.922	0.124	0.067	0.905	0.108	0.896	-	-	-
Bao et al. (2022)	0.928	0.122	-	0.885	0.120	0.869	-	-	-
Jin et al. (2022)	0.920	0.118	0.063	0.900	0.104	0.895	-	-	-
Gupta et al. (2022)	0.943	0.114	0.056	0.914	0.110	0.879	0.919	0.113	0.983
Miao et al. (2023)	0.934	0.123	0.065	0.917	0.109	0.908	-	-	-
Tafasca et al. (2023)	0.939	0.122	0.062	0.914	0.109	0.834	0.935	0.107	0.986
Tafasca et al. (2024)	0.944	0.113	0.057	-	0.107	0.891	-	0.106	0.990
Song et al. (2024)	0.949	0.105	0.047	0.938	0.102	0.905	-	-	-
Ryan et al. (2025)*	0.958	0.099	0.041	0.937	0.103	0.903	0.951	0.101	0.994
Gemini 3.5 Flash	-	0.128	0.067	-	0.138	-	-	0.111	-
PaGE ViT-S Distill	0.964	0.086	0.033	0.964	0.074	0.937	0.970	0.075	0.997
PaGE ViT-S+ Distill	0.965	0.086	0.033	0.965	0.074	0.939	0.970	0.075	**0.997**
PaGE ViT-B Distill	**0.966**	0.081	0.029	0.969	0.068	0.945	0.973	0.070	0.997
PaGE ViT-H+	0.966	**0.080**	**0.029**	**0.972**	**0.064**	**0.951**	**0.975**	**0.069**	0.995
Human	0.924	0.096	0.040	0.921	0.051	0.925	0.911	0.048	0.993

Table 3: Comparison between prior art, the PaGE family, and humans. All four PaGE models far outperform the previous SOTA, with PaGE ViT-H+ and PaGE ViT-B Distill achieving human-level performance. *Gaze-LLE ViT-L, results reported in the original paper.

We provide a full comparison with prior work on GazeFollow, VAT and ChildPlay in Table 3. We also include Gemini 3.5 Flash, a SOTA multimodal LLM in our comparison (see Appendix J for MLLM evaluation procedure). On GazeFollow, PaGE exceeded human performance on all three metrics. As each test sample in GazeFollow has around 10 human annotations, our results demonstrate that PaGE is better than individual humans at aligning with the *overall human consensus*. On VAT, PaGE outperformed humans on AUC and $AP_{in/out}$, while reducing the human-AI gap on L2 error by 75% from 0.052 (Gaze-LLE ViT-L) to 0.013 (PaGE ViT-H+). Similarly, on ChildPlay, PaGE outperformed humans on AUC and $AP_{in/out}$ while reducing the human-AI gap on L2 error by 60% from 0.053 (Gaze-LLE ViT-L) to 0.021 (PaGE ViT-H+). **We are the first to push these human-annotated benchmarks, where human performance is a soft ceiling, towards saturation.**

To show that PaGE generalizes across datasets, we evaluate it on GOO-Real (Tomas et al., 2021) and a benchmark emphasizing head and eye orientation (Zhang et al., 2026b) **without additional**

Model	GOO-Real AUC ↑	GOO-Real L2 ↓	Zhang et al. (2026b) Accuracy
Chong et al. (2020)	0.670*	0.334*	-
Miao et al. (2023)	0.869*	0.202*	-
Ryan et al. (2025)†	0.898*	0.175*	45.40
PaGE ViT-S Distill	0.914	0.151	72.42
PaGE ViT-S+ Distill	0.915	0.149	72.90
PaGE ViT-B Distill	0.914	0.153	**76.29**
PaGE ViT-H+ Teacher	**0.930**	**0.140**	71.98

Table 4: Results on GOO-Real and Zhang et al. (2026b)'s benchmark. On both datasets, PaGE achieved SOTA without training. *Results reported by Ryan et al. (2025). †Gaze-LLE ViT-L.

training. The results are in Table 4. The L2 error on GOO-Real is larger than that on other datasets since many images in GOO-Real have the person facing away from the camera. In these cases, humans also struggle to identify the gaze target. However, since Tomas et al. (2021) did not provide a human baseline, the human-AI gap cannot be measured. Meanwhile, on Zhang et al.'s benchmark, PaGE achieves SOTA performance. Since that benchmark is challenging for MLLMs (Zhang et al., 2026b), the result underscores PaGE's potential in "Gaze + MLLM" pipelines like GazeCoT (Ye et al., 2026). Qualitative results for all 5 datasets are in Figure 2.



Figure 2: Qualitative results of PaGE ViT-H+. Red dots and bounding boxes are ground truths.

4.2 Analysis of Finetuning and Feature Distillation

Experiment	GazeFollow AUC	GazeFollow Avg L2	GazeFollow Min L2	VideoAttentionTarget AUC	VideoAttentionTarget L2	VideoAttentionTarget AP	ChildPlay AUC	ChildPlay L2	ChildPlay AP
Train Decoder Only, no SFT	0.9599	0.0939	0.0380	0.9557	0.0886	0.9421	0.9652	0.0883	0.9961
Train Full Model (End to End)	0.9073	0.1881	0.1175	0.8795	0.1733	0.8449	0.9011	0.1632	0.9842
Train Decoder, then SFT	**0.9625**	**0.0892**	**0.0345**	**0.9618**	**0.0844**	**0.9491**	**0.9690**	**0.0810**	**0.9971**
Distill w/ 100k Samples + SFT	0.9639	0.0868	0.0333	0.9649	0.0724	0.9463	0.9703	0.0742	0.9945
Distill w/ 200k Samples + SFT	0.9647	0.0839	0.0316	0.9658	0.0718	**0.9484**	0.9710	0.0728	0.9958
Distill w/ 400k Samples + SFT	0.9651	0.0834	0.0307	0.9666	0.0721	0.9415	0.9713	0.0719	0.9965
Distill w/ 1.17M Samples + SFT	**0.9660**	**0.0814**	**0.0295**	**0.9688**	**0.0677**	0.9450	**0.9734**	**0.0697**	**0.9969**

Table 5: Ablation study for SFT and distillation. All experiments are done with PaGE ViT-B.

In this section, we ablate elements of SFT and feature distillation, which are central to our contribution to the training recipe, and analyze their performance impact. First, we show that training the full model only works when we have already trained a decent decoder on a frozen backbone. Training the full model from the start results in very poor performance (Table 5). We hypothesize

that a randomly initialized decoder unable to make sense of backbone features would create noisy gradients that hinder the backbone from learning useful adaptations on top of DINOv3. Second, we compare student models distilled with different amounts of unlabeled samples. More distillation data consistently leads to better performance. Together, these results validate our novel training recipe, which could also be used by future gaze estimation models. More analysis is in Appendix I.

4.3 You Can Teach an Old Dog New Tricks

To demonstrate that PaGE is limited by existing gaze datasets, we explore the specialized task of tracking gaze on a computer screen (Liu et al., 2026). We choose this task because it requires spatial perception and reasoning beyond human capabilities. We use data from 21 users for training, and 3 users for evaluation. We combine the train set used in previous sections (a mix of GazeFollow, VAT and ChildPlay, 200k images) with the new data (20k images) and finetune the PaGE ViT-B student on this expanded train set. We report results in Table 6.

	GazeFollow			VideoAttentionTarget			**Liu et al. (2026)**		
	AUC↑	Avg L2↓	Min L2↓	AUC↑	L2↓	APin/out↑	AUC↑	L2↓	APin/out↑
SFT w/o Screen Data	0.9660	0.0814	0.0295	0.9688	0.0677	0.9450	0.9562	0.0961	0.9978
SFT w/ Screen Data	0.9662	0.0806	0.0294	0.9685	0.0690	0.9375	0.9858	0.0524	0.9991
Performance Impact	+0.02%	-0.98%	-0.34%	-0.03%	+1.92%	-0.79%	+3.10%	**-45.47%**	+0.13%

Table 6: Including gaze-on-screen data when finetuning the student model significantly improves performance on that task (-45% L2 error) without impacting performance elsewhere too much.

The results show that the added data did not impact GazeFollow and VAT metrics by much, but significantly improved gaze tracking on screens. This is evident in Figure 3: without the additional data, the model simply guessed that the person was looking at the center of the screen; with additional training, however, the model learned to leverage nuanced details of the person's eye and head. **This demonstrates that PaGE still has considerable untapped model capacity. Diverse datasets with accurate ground truth are essential for future data-driven progress in gaze estimation.**

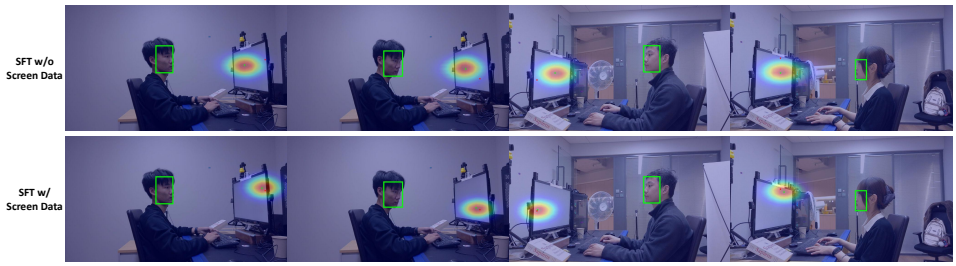

Figure 3: Qualitative results on Liu et al.'s dataset. Red dots are ground truth annotations.

4.4 Are We Evaluating Human-level Models Fairly?

Evaluation Protocol	GazeFollow Avg L2	GazeFollow Min L2	VideoAttentionTarget L2	ChildPlay L2
Biased	0.0804	0.0288	0.0643	0.0687
Unbiased	0.0786	0.0277	0.0624	0.0662
Human-AI Gap Reduction	-	-	-14.3% (0.0133 → 0.0114)	-12.1% (0.0207 → 0.0182)

Table 7: Removing the systematic bias in evaluation. No human-AI gap is reported for GazeFollow because PaGE ViT-H+ surpassed the human baseline on both GazeFollow L2 metrics.

PaGE nearly saturates existing metrics. At this point, the errors introduced by the evaluation procedure begin to matter. When calculating L2 errors, the standard evaluation procedure of VAT and ChildPlay derives the normalized coordinates of the model's predicted gaze target from a 64×64 heatmap, while humans annotate the images at native resolution. This creates room for *discretization errors*. Specifically, open-source evaluation scripts used in previous work (Chong et al., 2020;

Ryan et al., 2025) calculate the predicted normalized gaze target as $\hat{g} = (\frac{\hat{x}}{64}, \frac{\hat{y}}{64})$, introducing a shift towards the top left of the image. We correct for this bias and use $\hat{g} = (\frac{\hat{x}+0.5}{64}, \frac{\hat{y}+0.5}{64})$, and report the results in Table 7. It is remarkable that this simple error accounts for 12%-14% of the apparent L2 gap between PaGE and humans. Therefore, future datasets should use unbiased evaluation or higher resolution heatmaps (see Appendix L) for fairness. **Note that the results reported elsewhere in this paper still use the standard biased protocols to maintain comparability with prior work.**

4.5 Discussion: Is PaGE a Universal Visual Attention Estimator?

How far can generalization take us? We are pleasantly surprised to find that PaGE can estimate the gaze of *animals* to some degree. In fact, for the examples in Figure 4, our human intuition mostly agrees with PaGE's predictions across many different species. Even more surprisingly, we found that PaGE can predict which area a *camera* is filming. Part of this generalizability could plausibly be attributed to the vast train set of DINOv3. In addition to that, we hypothesize that PaGE has learned a type of ***universal representation of visual attention*** through training on human gaze data alone. This representation combines global features (e.g., social dynamics, saliency) and local ones (e.g., head and eye orientation, pose), and can partially generalize to animals and cameras. For example, the primary cue in the eagle-and-rodent image is the animals' eye orientation; the gorilla case, despite blurry faces, still has pose and social dynamics; the penguin example offers a strong pose cue (the animal is facing away from the camera). Meanwhile, we hypothesize that PaGE predicts camera targets primarily through pose and saliency cues.



Figure 4: Zero-shot visual attention estimation beyond real-world human gaze. PaGE can generalize to animals, cameras, and animation. Do you agree with PaGE's predictions?

Another type of generalization concerns *non-realistic images*. To this end, we test PaGE on animation frames with various styles (Figure 4), which often break the fine-grained local features (e.g., eye features) and strict spatial priors found in realistic images. While PaGE does encounter some challenges as expected (e.g., the wrong in/out prediction at the bottom-left of the *animation* column shows that PaGE has limited understanding of animated eyes), it still works reasonably well in many cases. This provides further qualitative evidence that PaGE has learned a universal, comprehensive representation of visual attention that still works in artificial images that break real-world priors.

5 Conclusion

We proposed PaGE, the first human-level gaze target estimation model. PaGE achieved SOTA performance and high efficiency through (1) a novel architecture designed to model complex interactions between scene and head features, (2) an improved training recipe that combines multiple datasets, decoder training, and full-scale finetuning, and (3) feature distillation on large-scale unlabeled data. We dissected how each design choice contributed to performance, explored expanding model capability to a new task, and designed fairer evaluation protocols. We ended by discussing PaGE's ability to generalize beyond human gaze and realistic images, paving the way for a *GazeAnything* model. We hope our work can turn a new *page* for the field of gaze target estimation and drive the development of better models, datasets, and downstream applications.

AI USE STATEMENT

(This section is **required** and does not count toward the page limit.)

In this work, we used generative AI tools to polish the implementation of some PyTorch modules. Note that the original implementation were done by humans, and the polished code was double-checked by at least 2 authors to ensure that the output remained consistent. We have not used generative AI tools for designing or providing feedback on research methodology or experiments, cleaning and reformatting datasets, interpreting results, or writing the paper. Other tasks requiring disclosure are not applicable to this work. These include generating synthetic datasets, helping develop theoretical models or conceptual frameworks, formulating mathematical claims, providing critical ingredients for proving mathematical claims, assisting in the writing of proofs, proposing or refining hypotheses, assisting with translation, and supporting qualitative and thematic data analysis. We have reviewed all AI-assisted work and take full responsibility for the final content of this work, including text, claims and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

(This section is **recommended** and does not count toward the page limit.)

We acknowledge that gaze target estimation poses privacy concerns in certain applications. Some of the concerns are discussed in detail by Ye et al. (2026). We firmly believe that the informed consent of both the user and bystanders is the prerequisite of the responsible and ethical deployment of gaze estimation algorithms. This work, in particular, follows the convention set by previous work on gaze target estimation (Tafasca et al., 2024; Ryan et al., 2025), and uses publicly available datasets for all experiments (i.e., no new experiments involving human subjects). The usage guidelines of the datasets were strictly followed. In addition, we aim to be fully transparent with our research, and will publicly release all code and model checkpoints. Therefore, we believe that **this work itself does not raise new ethical concerns**.

REPRODUCIBILITY STATEMENT

(This section is **recommended** and does not count toward the page limit.)

It is our aim to make this paper fully reproducible. The code for data processing, model training, and evaluation contain all of the details needed for reproducing the results, and are included in the supplementary materials. Detailed instructions can be found in the README file. As of the time of submission, all datasets used in this work are available on their respective official websites. In addition, Appendix A includes the hyperparameters for the main experiments.

REFERENCES

Henny Admoni and Brian Scassellati. Social eye gaze in human-robot interaction: a review. *Journal of Human-Robot Interaction*, 6(1):25–63, 2017.

Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, and Bernt Schiele. 2d human pose estimation: New benchmark and state of the art analysis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.

Jun Bao, Buyu Liu, and Jun Yu. Escnet: Gaze target detection with the understanding of 3d scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14126–14135, June 2022.

Bingyi Cao, Koert Chen, Kevis-Kokitsi Maninis, Kaifeng Chen, Arjun Karpur, Ye Xia, Sahil Dua, Tanmaya Dabral, Guangxing Han, Bohyung Han, Joshua Ainslie, Alex Bewley, Mithun Jacob, René Wagner, Washington Ramos, Krzysztof Choromanski, Mojtaba Seyedhosseini, Howard Zhou, and Andre Araujo. Tipsv2: Advancing vision-language pretraining with enhanced patch-text alignment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 29325–29335, June 2026a.

Xu Cao, Houze Yang, Vipin Gunda, Zhongyi Zhou, Tianyu Xu, Adarsh Kowdle, Inki Kim, and James M. Rehg. Gaze target estimation anywhere with concepts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 31304–31315, June 2026b.

Zhe Chen, Yuchen Duan, Wenhai Wang, Junjun He, Tong Lu, Jifeng Dai, and Yu Qiao. Vision transformer adapter for dense predictions. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=plKu2GByCNW.

Eunji Chong, Nataniel Ruiz, Yongxin Wang, Yun Zhang, Agata Rozga, and James M. Rehg. Connecting gaze, scene, and attention: Generalized attention estimation via joint modeling of gaze and scene saliency. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

Eunji Chong, Yongxin Wang, Nataniel Ruiz, and James M. Rehg. Detecting attended visual targets in video. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

Yi Fang, Jiapeng Tang, Wang Shen, Wei Shen, Xiao Gu, Li Song, and Guangtao Zhai. Dual attention guided gaze target detection in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11390–11399, June 2021.

Logan Frank and Jim Davis. What makes a good dataset for knowledge distillation? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 23755–23764, June 2025.

Anshul Gupta, Samy Tafasca, and Jean-Marc 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 (CVPR) Workshops*, pp. 5041–5050, June 2022.

Byeongho Heo, Song Park, Dongyoon Han, and Sangdoo Yun. Rotary position embedding for vision transformer. In Aleš Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol (eds.), *Computer Vision – ECCV 2024*, pp. 289–305, Cham, 2025. Springer Nature Switzerland. ISBN 978-3-031-72684-2.

Zhengxi Hu, Yuxue Yang, Xiaolin Zhai, Dingye Yang, Bohan Zhou, and Jingtai Liu. Gfie: A dataset and baseline for gaze-following from 2d to 3d in indoor environments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8907–8916, June 2023.

Tianlei Jin, Qizhi Yu, Shiqiang Zhu, Zheyuan Lin, Jie Ren, Yuanhai Zhou, and Wei Song. Depth-aware gaze-following via auxiliary networks for robotics. *Engineering Applications of Artificial Intelligence*, 113:104924, 2022. ISSN 0952-1976. doi: https://doi.org/10.1016/j.engappai.2022.104924. URL https://www.sciencedirect.com/science/article/pii/S0952197622001464.

Ivan Krasin, Tom Duerig, Neil Alldrin, Vittorio Ferrari, Sami Abu-El-Haija, Alina Kuznetsova, Hassan Rom, Jasper Uijlings, Stefan Popov, Shahab Kamali, Matteo Malloci, Jordi Pont-Tuset, Andreas Veit, Serge Belongie, Victor Gomes, Abhinav Gupta, Chen Sun, Gal Chechik, David Cai, Zheyun Feng, Dhyanesh Narayanan, and Kevin Murphy. Openimages: A public dataset for large-scale multi-label and multi-class image classification. *Dataset available from https://storage.googleapis.com/openimages/web/index.html*, 2017.

Dongze Lian, Zehao Yu, and Shenghua Gao. Believe it or not, we know what you are looking at! In C. V. Jawahar, Hongdong Li, Greg Mori, and Konrad Schindler (eds.), *Computer Vision – ACCV 2018*, pp. 35–50, Cham, 2019. Springer International Publishing. ISBN 978-3-030-20893-6.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars (eds.), *Computer Vision – ECCV 2014*, pp. 740–755, Cham, 2014. Springer International Publishing. ISBN 978-3-319-10602-1.

Chang Liu, Jiaqi Liu, Zhoutong Ye, Xinjie Shen, Chun Yu, and Yuanchun Shi. Aa: A multi-view multimodal dataset for screen-based gaze estimation, 2026. URL https://arxiv.org/abs/2606.31211.

Qiaomu Miao, Minh Hoai, and Dimitris Samaras. Patch-level gaze distribution prediction for gaze following. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 880–889, January 2023.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel HAZIZA, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=a68SUt6zFt. Featured Certification.

Matej Palider, Omar Eldardeer, and Viktor Kocur. Gaze estimation for human-robot interaction: Analysis using the nico platform, 2025. URL https://arxiv.org/abs/2509.24001.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/radford21a.html.

Adria Recasens, Aditya Khosla, Carl Vondrick, and Antonio Torralba. Where are they looking? In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015. URL https://proceedings.neurips.cc/paper_files/paper/2015/file/ec8956637a99787bd197eacd77acce5e-Paper.pdf.

Fiona Ryan, Ajay Bati, Sangmin Lee, Daniel Bolya, Judy Hoffman, and James M. Rehg. Gazelle: Gaze target estimation via large-scale learned encoders. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 28874–28884, June 2025.

Oriane Siméoni, Huy V Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, et al. Dinov3. *arXiv preprint arXiv:2508.10104*, 2025.

Yuehao Song, Xinggang Wang, Jingfeng Yao, Wenyu Liu, Jinglin Zhang, and Xiangmin Xu. ViTGaze: Gaze following with interaction features in vision transformers. *Visual Intelligence*, 2(1):31, 2024. ISSN 2731-9008. doi: 10.1007/s44267-024-00064-9. URL https://doi.org/10.1007/s44267-024-00064-9.

Samy Tafasca, Anshul Gupta, and Jean-Marc Odobez. Childplay: A new benchmark for understanding children's gaze behaviour. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 20935–20946, October 2023.

Samy Tafasca, Anshul Gupta, and Jean-Marc Odobez. Sharingan: A transformer architecture for multi-person gaze following. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2008–2017, June 2024.

Henri Tomas, Marcus Reyes, Raimarc Dionido, Mark Ty, Jonric Mirando, Joel Casimiro, Rowel Atienza, and Richard Guinto. Goo: A dataset for gaze object prediction in retail environments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pp. 3125–3133, June 2021.

Francesco Tonini, Nicola Dall'Asen, Cigdem Beyan, and Elisa Ricci. Object-aware gaze target detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 21860–21869, October 2023.

Danyang Tu, Xiongkuo Min, Huiyu Duan, Guodong Guo, Guangtao Zhai, and Wei Shen. End-to-end human-gaze-target detection with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2202–2210, June 2022.

Danyang Tu, Wei Shen, Wei Sun, Xiongkuo Min, and Guangtao Zhai. Joint gaze-location and gaze-object detection, 2023. URL https://arxiv.org/abs/2308.13857.

Zhoutong Ye, Xutong Wang, Chengwen Zhang, Ruiwen Zhang, Mingze Sun, Qinwei Li, Chun Yu, and Yuanchun Shi. Gazecot: Unleashing social intelligence in multimodal llms with gaze-informed chain-of-thought reasoning. In *Proceedings of the 2026 CHI Conference on Human Factors in Computing Systems*, CHI '26, New York, NY, USA, 2026. Association for Computing Machinery. ISBN 9798400722783. doi: 10.1145/3772318.3790922. URL https://doi.org/10.1145/3772318.3790922.

Chengwen Zhang, Chun Yu, Borong Zhuang, Haopeng Jin, Qingyang Wan, Zhuojun Li, Zhe He, Zhoutong Ye, Yu Mei, Chang Liu, Weinan Shi, and Yuanchun Shi. Hisync: Spatio-temporally aligning hand motion from wearable imu and on-robot camera for command source identification in long-range hri. In *Proceedings of the 2026 CHI Conference on Human Factors in Computing Systems*, CHI '26, New York, NY, USA, 2026a. Association for Computing Machinery. ISBN 9798400722783. doi: 10.1145/3772318.3790345. URL https://doi.org/10.1145/3772318.3790345.

Zory Zhang, Pinyuan Feng, Bingyang Wang, Tianwei Zhao, Suyang Yu, Qingying Gao, Hokin Deng, Ziqiao Ma, Yijiang Li, and Dezhi Luo. Vision-language models mistake head orientation for gaze direction: Nonverbal conversation cues, 2026b. URL https://arxiv.org/abs/2506.05412.

A Detailed Training Procedures

We provide the technical details of how we trained the teacher and student models. All training stages use the same data augmentation pipeline as Gaze-LLE's (random cropping, horizontal flipping, and bounding box jittering). To speed up training, we train all models in BF16 using PyTorch's Automatic Mixed Precision (AMP) training. The models were trained on a single Nvidia H100 GPU. Finetuning / distilling ViT-L and ViT-H+ models requires much more VRAM than the 80GB available, so we split the batches into smaller ones (30 for ViT-L, 12 for ViT-H+), and accumulate the gradient to achieve an effective batch size of 60.

A.1 Training the Teacher Model

As laid out in Section 3.2, we train the teacher model in 2 stages. We first train the decoder only and then finetune the full model (backbone included).

Stage 1: Decoder Training. The decoder is trained using AdamW with a weight decay of 0.05 and a 100-iteration linear LR warmup from $\eta_0 = 10^{-4}$. The rest of the recipe is the same as Gaze-LLE's: we train the model for 15 epochs with a batch size of 60 and an initial learning rate of $\eta = 10^{-3}$ after warmup, and use a cosine LR scheduler with $\eta_{min} = 10^{-7}$.

Stage 2: Finetuning. In this stage, the full model is trained with the AdamW optimizer for 5 epochs with a weight decay of 0.01 and a cosine LR schedule with $\eta_{min} = 10^{-7}$. We observe that this process is susceptible to overfitting and training instability. Therefore, we use a long 500-iteration linear warmup from $\eta_0 = 10^{-7}$, a low post-warmup initial learning rate of $\eta = 10^{-5}$, and a more aggressive $p = 0.5$ channel dropout for backbone features.

A.2 Distillation

Similar to the teacher model, we train the student model in two stages as well.

Stage 1: Feature Distillation. We train the student model with AdamW, using a weight decay of 0.01, an initial LR of $\eta = 2 \times 10^{-4}$ and a cosine LR schedule with $\eta_{min} = 10^{-7}$. The distillation runs for 20 epochs (14 more than what is needed to make a good vodka;).

Stage 2: Finetuning. Since the student only learned feature representation and not downstream prediction during distillation, we adopt a supervised finetuning regime on the labeled train set to fully adapt the student to the downstream task. Specifically, we initialize the student heatmap and in/out heads with teacher weights and finetune the model for 3 epochs. We use a post-warmup initial LR of $\eta = 2 \times 10^{-5}$. The rest of the finetuning regime is identical to the one in Appendix A.1.

B Impact of VAT and ChildPlay Frame Sampling

Experiment	VAT Frames	ChildPlay Frames	GazeFollow AUC	GazeFollow Avg L2	GazeFollow Min L2	VideoAttentionTarget AUC	VideoAttentionTarget L2	VideoAttentionTarget $AP_{in/out}$	ChildPlay AUC	ChildPlay L2	ChildPlay $AP_{in/out}$
GazeFollow-only	Not Used	Not Used	0.9595	0.0958	0.0391	0.9488	0.0870	N/A	0.9572	0.0963	N/A
Labeled Train Set (Main Experiment)	1/3	1/6	**0.9599**	0.0939	0.0380	0.9557	0.0886	**0.9421**	**0.9652**	**0.0883**	**0.9961**
Alternative VAT Sampling Intervals	All	1/6	0.9594	**0.0928**	**0.0373**	0.9552	0.0889	0.9360	0.9627	0.0942	0.9941
	1/6	1/6	0.9598	0.0949	0.0391	0.9542	0.0900	0.9345	0.9647	0.0902	0.9950
	1/12	1/6	0.9595	0.0956	0.0384	0.9544	**0.0859**	0.9246	0.9634	0.0924	0.9947
Alternative ChildPlay Sampling Intervals	1/3	All	0.9595	0.0946	0.0385	**0.9577**	0.0898	0.9305	0.9628	0.0954	0.9926
	1/3	1/3	0.9596	0.0965	0.0402	0.9547	0.0906	0.9342	0.9627	0.0960	0.9929
	1/3	1/12	0.9598	0.0955	0.0390	0.9570	0.0882	0.9332	0.9634	0.0910	0.9952

Table 8: The impact of different train set composition. Best performance is **bolded** while the runner-up is underlined. The composition we chose for our labeled train set (all GazeFollow samples + 1 out of every 3 VAT frames + 1 out of every 6 ChildPlay frames) yields the best overall result.

We conducted an ablation study on the composition of the labeled train set to validate our decision to sample VAT and ChildPlay frames at a reduced rate (Section 3.2.1). We train PaGE ViT-B (w/o finetuning) on a variety of data mixtures and report the results in Table 8. Sampling 1 out of 3 VAT frames and 1 out of 6 ChildPlay frames yields the best results. As a result, we choose that as the default composition of our labeled train set.

C Alternative Backbones

We explored using different backbones for PaGE. Specifically, we compare 4 ViT-L backbones—DINOv2, CLIP, TIPSv2, and DINOv3. We train PaGE ViT-L (w/o finetuning) using each of these backbones and report the results in Table 9. PaGE is capable of SOTA performance regardless of which backbone is used, although DINOv3 performed the best. This portability across backbones makes it likely that PaGE can continue to benefit from progress in general-purpose vision backbones.

Backbone	GazeFollow AUC	GazeFollow Avg L2	GazeFollow Min L2	VideoAttentionTarget AUC	VideoAttentionTarget L2	VideoAttentionTarget $AP_{in/out}$	ChildPlay AUC	ChildPlay L2	ChildPlay $AP_{in/out}$
DINOv2 (Oquab et al., 2024)	0.9606	0.0895	0.0361	0.9586	0.0856	0.9260	0.9673	0.0842	0.9956
CLIP (Radford et al., 2021)	0.9604	0.0910	0.0373	0.9625	0.0837	0.9298	0.9667	0.0859	**0.9966**
TIPSv2 (Cao et al., 2026a)	0.9606	0.0925	0.0361	0.9529	0.0920	0.9386	0.9648	0.0850	0.9958
DINOv3 (Siméoni et al., 2025)	**0.9623**	**0.0881**	**0.0338**	**0.9643**	**0.0804**	**0.9449**	**0.9678**	**0.0798**	0.9950
DINOv3 + CLIP	0.9620	0.0878	0.0339	0.9652	0.0801	0.9368	0.9690	0.0796	0.9957

Table 9: Performance of PaGE with different ViT-L backbones. All models were trained on the labeled train set without finetuning.

Considering that gaze is inherently social and is extensively described in text, we hypothesize that models trained on vision language (VL) tasks like CLIP and TIPSv2 could provide additional features that complement DINOv3. We further explore concatenating DINOv3 and CLIP feature maps along the *channel* dimension as the input to a PaGE decoder. Specifically, we use 512×512 scene input for DINOv3 ViT-L (16×16 patches) and 448×448 scene input for CLIP ViT-L (14×14 patches). This generates two $32 \times 32 \times 768$ feature maps that can be directly concatenated to form one $32 \times 32 \times 1536$ feature map. Similarly, we resize the head crop to 256×256 for DINOv3 and 224×224 for CLIP, and concatenate the head feature maps along the channel dimension.

We report the performance of this hybrid backbone in Table 9. The hybrid backbone slightly improves performance but has $2\times$ the computational cost, so we did not use the hybrid backbone in our main experiments. As VL backbones continue to evolve, we could expect hybrid backbones to bring larger improvements to gaze estimation that justify the extra FLOPs in the near future. We leave such explorations to future work.

D 2D RoPE for Cross Attention with Unified Coordinates

We provide the formal definition of 2D RoPE with unified global-local coordinates used in SIM's cross attention where we model the interaction between scene branch and head branch features. Let a patch token at grid position $p_n = (y_n, x_n)$. For a head with rotary dimension d_r, assume d_r is divisible by 4, and write the rotary part of each query/key as

$$q_n = (q_n^y, q_n^x, q_n^\perp), \qquad k_n = (k_n^y, k_n^x, k_n^\perp),$$

where $q_n^y, q_n^x, k_n^y, k_n^x \in \mathbb{C}^{d_r/4}$, using adjacent real channels as complex numbers. Define

$$\theta_i = \theta^{-i/(d_r/4)}, \qquad i = 0, \ldots, d_r/4 - 1.$$

Following Heo et al. (2025), we use $\theta = 100$ by default. See Appendix F.4 for results with alternative θ values.

For self-attention, axial 2D RoPE applies

$$\widetilde{q}_{n,i}^y = q_{n,i}^y e^{\mathrm{i}\theta_i y_n}, \qquad \widetilde{q}_{n,i}^x = q_{n,i}^x e^{\mathrm{i}\theta_i x_n},$$
$$\widetilde{k}_{n,i}^y = k_{n,i}^y e^{\mathrm{i}\theta_i y_n}, \qquad \widetilde{k}_{n,i}^x = k_{n,i}^x e^{\mathrm{i}\theta_i x_n}.$$

Then attention is computed as

$$\mathrm{Attn}(Q, K, V) = \mathrm{softmax}\left(\frac{\widetilde{Q}\widetilde{K}^\top}{\sqrt{d_h}}\right) V.$$

For cross-attention, let the global scene grid be $H_g \times W_g$, and let the local head crop grid be $H_\ell \times W_\ell$. Given a head bounding box

$$x_{\mathrm{bbox}} = (y_0, x_0, y_1, x_1) \in \mathbb{R}^4$$

in global-grid coordinates, assign local patch (a, b) the continuous global coordinate

$$\phi_\ell(a, b) = \left(y_0 + \frac{a + \frac{1}{2}}{H_\ell}(y_1 - y_0) - \frac{1}{2},\ x_0 + \frac{b + \frac{1}{2}}{W_\ell}(x_1 - x_0) - \frac{1}{2}\right).$$

Global-scene patches use their native coordinates

$$\phi_g(a, b) = (a, b).$$

For cross-attention from stream A to stream B, rotate queries using ϕ_A, rotate keys using ϕ_B, and leave values unrotated:

$$\mathrm{CrossAttn}(X_A, X_B) = \mathrm{softmax}\left(\frac{\widetilde{Q}_A \widetilde{K}_B^\top}{\sqrt{d_h}}\right) V_B.$$

Thus, for local-to-global attention, we use $\phi_A = \phi_\ell$, $\phi_B = \phi_g$. For global-to-local attention, we use $\phi_A = \phi_g$, $\phi_B = \phi_\ell$. This puts both branches in a unified coordinate system and implicitly encodes head location, allowing us to remove Gaze-LLE's learnable head prompt.

E Alternative SIM Layer Designs

We explored two other architectures (Figure 5) for modeling the interaction between the scene and head branches. We train a PaGE ViT-B (w/o finetuning) with each of the designs and report the results in Table 10. Coincidentally, Variant A is architecturally similar to ViTAdapter (Chen et al., 2023), which uses cross attention to model the interaction between a feature pyramid derived from convolutions and ViT features for dense prediction tasks. Our eventual SIM architecture, which includes an additional self attention for the head branch, leads to the best overall results.

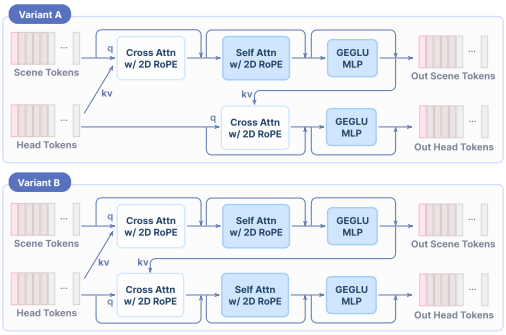

Figure 5: Two other SIM variants we explored. Both performed worse than our eventual architecture (see Figure 1).

Experiment	GazeFollow			VideoAttentionTarget			ChildPlay		
	AUC	Avg L2	Min L2	AUC	L2	APin/out	AUC	L2	APin/out
Variant A	0.9595	0.0971	0.0402	0.9554	0.0883	0.9350	0.9617	0.0942	0.9952
Variant B	0.9593	0.0962	0.0396	**0.9562**	**0.0875**	0.9277	0.9623	0.0981	0.9925
SIM	**0.9599**	**0.0939**	**0.0380**	0.9557	0.0886	**0.9421**	**0.9652**	**0.0883**	**0.9961**

Table 10: Comparison between SIM and two variants.

F More Ablation Studies of the PaGE Architecture

We conduct additional ablation studies on decoder hyperparameters and explore different model depths, token dimensions, FFN MLP ratios, FFN types, backbone feature dropout p, 2D RoPE θ, and learning rates. We run the experiments with PaGE ViT-B (w/o finetuning).

F.1 Model Depth and Width

We report the performance of PaGE models with varying depth, token dimension, and FFN MLP ratio in Tables 11, 12 and 13. The final combination of these hyperparameters we used in our paper represents a balance between different performance metrics and consistency with prior work.

# SIM Layers	GazeFollow			VideoAttentionTarget			ChildPlay		
	AUC	Avg L2	Min L2	AUC	L2	APin/out	AUC	L2	APin/out
3 Layers	0.9598	0.0956	0.0389	0.9543	0.0882	0.9285	0.9644	**0.0875**	0.9938
4 Layers	0.9597	0.0944	0.0381	0.9557	**0.0848**	0.9256	0.9639	0.0927	0.9932
5 Layers	**0.9599**	0.0939	**0.0380**	0.9557	0.0886	**0.9421**	**0.9652**	0.0883	**0.9961**
6 Layers	**0.9599**	0.0957	0.0386	**0.9564**	0.0883	0.9259	0.9622	0.0919	0.9949
7 Layers	**0.9599**	**0.0935**	0.0383	0.9560	0.0855	0.9390	0.9627	0.0941	0.9930

Table 11: PaGE performance with different numbers of SIM layers. 5 layers leads to the best overall performance, and is the default in this paper.

Token Dim	GazeFollow			VideoAttentionTarget			ChildPlay		
	AUC	Avg L2	Min L2	AUC	L2	APin/out	AUC	L2	APin/out
128	**0.9601**	**0.0933**	0.0384	**0.9587**	**0.0861**	0.9408	0.9649	**0.0882**	0.9950
192	0.9595	0.0952	0.0392	0.9580	**0.0861**	0.9374	0.9642	0.0888	0.9945
256	0.9599	0.0939	**0.0380**	0.9557	0.0886	**0.9421**	**0.9652**	0.0883	**0.9961**
384	0.9590	0.1003	0.0421	0.9490	0.0987	0.9267	0.9596	0.0965	0.9944

Table 12: PaGE performance with different token dimensions. While there is no significant difference between 128 and 256, we decided to use 256 to stay consistent with Ryan et al. (2025).

MLP Ratio	GazeFollow			VideoAttentionTarget			ChildPlay		
	AUC	Avg L2	Min L2	AUC	L2	AP	AUC	L2	AP
2	**0.9601**	0.0950	0.0389	0.9559	0.0869	0.9235	0.9644	0.0889	0.9956
4	0.9599	**0.0939**	**0.0380**	0.9557	0.0886	**0.9421**	**0.9652**	**0.0883**	**0.9961**
6	0.9598	0.0943	0.0381	**0.9588**	**0.0806**	0.9238	0.9630	0.0928	0.9937
8	**0.9601**	0.0954	0.0390	0.9554	0.0880	0.9374	0.9626	0.0934	0.9945

Table 13: PaGE performance with different MLP ratios. A MLP ratio of 4 leads to the best overall results.

F.2 GEGLU vs SwiGLU

We report the performance when we substitute GEGLU MLP with SwiGLU MLP in PaGE in Table 14. GEGLU performs better than SwiGLU, so we use GEGLU as the default in PaGE.

MLP Layer	GazeFollow			VideoAttentionTarget			ChildPlay		
	AUC	Avg L2	Min L2	AUC	L2	AP	AUC	L2	AP
SwiGLU	0.9595	0.0943	0.0386	**0.9582**	**0.0857**	0.9364	0.9633	0.0935	0.9950
GEGLU	**0.9599**	**0.0939**	**0.0380**	0.9557	0.0886	**0.9421**	**0.9652**	**0.0883**	**0.9961**

Table 14: PaGE with different types of MLP layers. GEGLU has the best overall performance and is used as the default.

F.3 DINO Feature Dropout Rates

We report the impact of different DINO feature dropout probabilities (p) in Table 15. We settled on $p = 0.1$ because it led to the best overall results.

DINO Feature Dropout p	GazeFollow			VideoAttentionTarget			ChildPlay		
	AUC	Avg L2	Min L2	AUC	L2	AP	AUC	L2	AP
0.0	0.9580	0.0963	0.0403	0.9564	0.0889	0.9264	0.9644	0.0933	0.9943
0.1	0.9599	**0.0939**	**0.0380**	0.9557	0.0886	**0.9421**	**0.9652**	**0.0883**	**0.9961**
0.3	0.9604	0.0949	0.0381	**0.9566**	**0.0856**	0.9350	0.9640	0.0917	0.9957
0.5	**0.9609**	0.0953	0.0382	0.9540	0.0873	0.9336	0.9621	0.0918	0.9950

Table 15: Impact of DINO feature dropout p.

F.4 2D RoPE Frequency

θ controls the frequency of 2D RoPE. Both Heo et al. (2025) and DINOv3 (Siméoni et al., 2025) use $\theta = 100$, which is also the default in our paper. We further explore $\theta = 10$ and $\theta = 10000$ and report the results in Table 16.

RoPE θ	GazeFollow			VideoAttentionTarget			ChildPlay		
	AUC	Avg L2	Min L2	AUC	L2	$AP_{in/out}$	AUC	L2	$AP_{in/out}$
10	0.9594	0.0948	0.0387	0.9552	**0.0878**	0.9321	0.9650	0.0885	0.9954
100	**0.9599**	**0.0939**	0.0380	0.9557	0.0886	**0.9421**	**0.9652**	**0.0883**	**0.9961**
10000	0.9595	0.0940	**0.0379**	**0.9583**	0.0882	0.9302	0.9628	0.0922	0.9950

Table 16: Effect of different 2D RoPE θ.

F.5 Learning Rate

We explore the effect of different learning rates and present the results in Table 17. We encountered several loss spikes when training the model with $\eta = 3 \times 10^{-3}$, which could be responsible for the abnormally low performance, especially on $AP_{in/out}$ where the model performed similar to blindly guessing "in frame" for all samples.

LR	GazeFollow			VideoAttentionTarget			ChildPlay		
	AUC	Avg L2	Min L2	AUC	L2	APin/out	AUC	L2	APin/out
3e-3	0.9486	0.1238	0.0602	0.9218	0.1247	0.6459	0.9419	0.1104	0.9444
1e-3	**0.9599**	**0.0939**	**0.0380**	**0.9557**	0.0886	**0.9421**	**0.9652**	**0.0883**	**0.9961**
3e-4	0.9577	0.0954	0.0398	0.9555	**0.0857**	0.9329	0.9637	0.0934	0.9945
1e-4	0.9573	0.0971	0.0409	0.9552	0.0923	0.9179	0.9630	0.0940	0.9937

Table 17: Effect of different learning rates.

G Distillation Set Data Quality

As laid out in Section 3.3.1, the distillation set obtains head bounding boxes from a YOLO head detector. This automated approach has been validated in previous work (Ryan et al., 2025). We further confirm that the quality of our unlabeled distillation set is on par with the labeled train set through a simple experiment. Since the labeled train set has $200k$ images, we distill 2 student models using the labeled train set and a $200k$ subset of the distillation set, respectively. Both models are then finetuned on the labeled train set per Section A.2. We report the results in Table 18. Both yielded highly similar results, suggesting that our unlabeled distillation set is both high-quality and scalable.

Distillation Data	GazeFollow			VideoAttentionTarget			ChildPlay		
	AUC	Avg L2	Min L2	AUC	L2	APin/out	AUC	L2	APin/out
Labeled Train Set (200k)	0.9644	0.0830	0.0312	0.9677	0.0720	0.9481	0.9730	0.0734	0.9947
200k Unlabeled	0.9647	0.0839	0.0316	0.9658	0.0718	0.9484	0.9710	0.0728	0.9958

Table 18: Distillation with different data. The difference in performance is negligible.

H Profiling PaGE ViT-S Distill for Practical Deployment

In Section 3.3.2, we reported the FLOPs of all PaGE variants, with the distilled models retaining SOTA performance while being lightweight. To further demonstrate that models like PaGE ViT-S are suitable for practical use cases involving robots or edge devices (Admoni & Scassellati, 2017; Zhang et al., 2026a; Palider et al., 2025), we profile the model on an Nvidia RTX 4090 GPU and an Intel i9-14900K CPU. GPU inference took on average 16ms per frame, while CPU-only inference took around 190ms. This result is achieved using BF16 (PyTorch AMP), which is in line with the training and evaluation procedures used elsewhere in this paper.

I Further Analysis of SFT and Distillation

First, we ablate our distillation objective and present the results in Table 19. Adding the cosine term and auxiliary loss affects GazeFollow positively and VAT negatively, while the impact on ChildPlay is minimal. Considering GazeFollow is much more diverse and reflective of downstream applications, we decided to keep the cosine term and auxiliary loss. Meanwhile, using L1 instead of the more common MSE loss brings clear improvements.

Experiment	GazeFollow			VideoAttentionTarget			ChildPlay		
	AUC	Avg L2	Min L2	AUC	L2	AP	AUC	L2	AP
No Cosine Term (200k)	0.9646	0.0849	0.0323	**0.9671**	**0.0684**	0.9459	**0.9714**	0.0731	**0.9961**
L1 Term → MSE Term (200k)	0.9603	0.0946	0.0388	0.9609	0.0793	0.9408	0.9674	0.0790	0.9952
No Auxiliary Loss (200k)	0.9643	0.0848	0.0321	0.9662	0.0696	0.9426	0.9713	**0.0724**	0.9954
$\mathcal{L}_{distill}$ (200k)	**0.9647**	**0.0839**	**0.0316**	0.9658	0.0718	**0.9484**	0.9710	0.0728	0.9958

Table 19: Ablation study for the distillation objective. All experiments are done with PaGE ViT-B.

We compare the performance of the student model before and after SFT. The before-SFT results were obtained by directly using the teacher model's heatmap and in/out heads on top of the student's final layer features. The results in Table 20 show that post-distillation SFT has an overall positive impact on performance, especially on ChildPlay.

Experiment	GazeFollow			VideoAttentionTarget			ChildPlay		
	AUC	Avg L2	Min L2	AUC	L2	AP	AUC	L2	AP
Distillation Only	0.9657	0.0817	0.0299	**0.9702**	0.0680	**0.9493**	0.9732	0.0722	**0.9970**
Distillation + SFT	**0.9660**	**0.0814**	**0.0295**	0.9688	**0.0677**	0.9450	**0.9734**	**0.0697**	0.9969

Table 20: SFT improves overall performance of the distilled ViT-B student model.

J Evaluating Gemini 3.5 Flash

In this section, we present the detailed evaluation procedure that led to the Gemini 3.5 Flash results reported in Table 3. We formulate the evaluation task as a VQA task with structured output. We overlay the scene image with the bounding box of the person's head, which acts like a visual prompt. Apart from the image, we also provide the model with the following text prompt:

```
Task:
1. Look at the whole image and the specified person's head box.
2. The tested person's head is marked by a red rectangle in the image.
3. Decide whether this person's gaze target is inside the image.
4. If it is inside, output one best gaze target point.
Coordinate system:
- Use a 1000x1000 grid over the image that you see.
- (0, 0) is the top-left corner and (999, 999) is the bottom-right corner.
- The image sent to you is {sent_width}x{sent_height} pixels.
- Head box on the 1000x1000 grid is [xmin, ymin, xmax, ymax] = bbox.
Return JSON only, with this exact shape: {"inout":"in","x":123,"y":456}
Rules:
- "inout" must be either "in" or "out".
- If "inout" is "out", set "x" and "y" to null.
- If "inout" is "in", "x" and "y" must be integers from 0 to 999.
- Do not return explanations, Markdown, or extra keys.
```

The 1000×1000 grid coordinate system is in line with Gemini's official documentation. We convert that coordinate into actual coordinates on the image during evaluation. Furthermore, this procedure yields a binary "in" or "out" classification for in/out prediction. This means that both $AP_{in/out}$ (works well with a probabilistic output) and AUC (designed to evaluate heatmaps) are unsuitable for evaluating Gemini. In contrast, only L2 results are comparable between gaze estimation models and Gemini, so we only report L2 metrics in Section 4.1. We further report the F1 scores for in/out prediction for both Gemini and PaGE ViT-H+, which (unlike $AP_{in/out}$) is directly comparable. Gemini achieved an F1 score of 0.811 on VAT and 0.973 on ChildPlay. PaGE ViT-H+ achieved an F1 score of 0.897 on VAT and 0.985 on ChildPlay, outperforming Gemini.

K More Qualitative Examples

We first present cases where PaGE failed in Figure 6. Many of these cases have occluded faces and eyes, while others depict moments of *gaze shifts*. Blurry images also pose challenges. We expect the model to perform better in at least some of the cases if given enough training data.

In addition, we observe that, in rare cases, PaGE is sensitive to small variations in the head bounding box. We present an example in Figure 7. While we find these cases to be very infrequent, future work should focus on mitigating this sensitivity, probably through stronger data augmentation.

In Figure 8, we present some cases where the human annotation is unreliable and PaGE's prediction makes more sense. This underscores the importance of future datasets providing actual ground truth instead of relying on human annotators.

L High Resolution Heatmap Output

As discussed in Section 4.4, the discretization errors stemming from the low resolution 64×64 heatmap used in standard evaluation protocols are unfair to gaze estimation models when com-

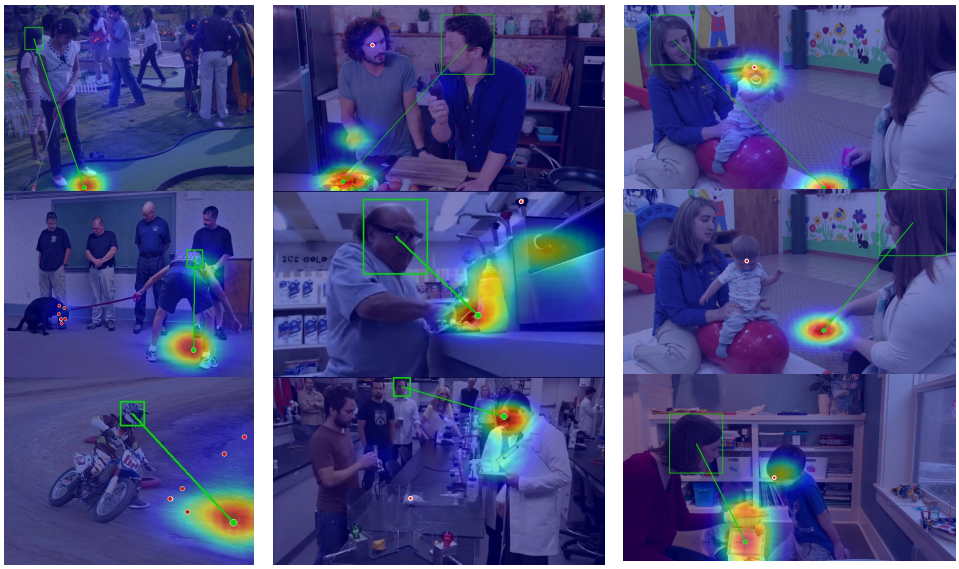

Figure 6: Failure cases where PaGE performed poorly.

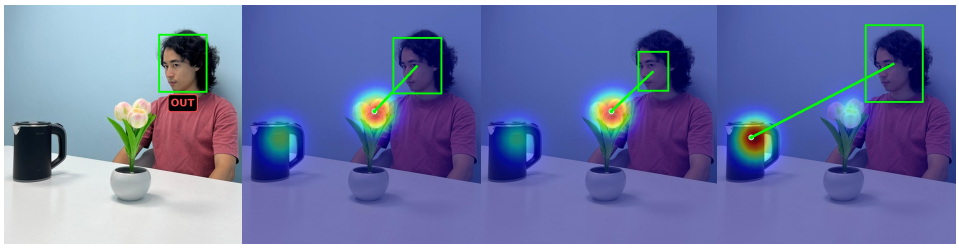

Figure 7: A case from Zhang et al. (2026b)'s dataset sensitive to changes in the head bounding box. All 4 bounding boxes shown here are plausible and unambiguous, but lead to different predictions.



Figure 8: GazeFollow test split samples with incorrect annotations. A common theme is annotators mistakenly labeling the gaze target of an individual unrelated to the head bounding box.

paring their performance to that of humans. While the fix we introduced in Section 4.4 removed a systematic bias towards the top left of the image, the underlying cause, discretization, remains. Here, we further explore the performance impact of training a PaGE ViT-H+ to output 512×512 heatmaps that cause negligible discretization errors.

We replicate the training regime for the PaGE ViT-H+ teacher model, but this time the model outputs 512×512 heatmaps, and we use 512×512 ground truth heatmaps as supervision. The increased resolution causes training instabilities, so we use a lower 3×10^{-4} learning rate instead of 10^{-3}. The SFT learning rate remains the same. We report the results in Table 21. Overall, using high resolution 512×512 outputs seems like a valid way to reduce systematic error in evaluation.

Heatmap Resolution	GazeFollow AUC	GazeFollow Avg L2	GazeFollow Min L2	VideoAttentionTarget AUC	VideoAttentionTarget L2	VideoAttentionTarget $AP_{in/out}$	ChildPlay AUC	ChildPlay L2	ChildPlay $AP_{in/out}$
64×64 Biased	0.9659	0.0804	0.0288	0.9719	0.0643	**0.9509**	0.9746	0.0687	0.9954
64×64 Unbiased	0.9659	**0.0786**	0.0277	0.9719	**0.0624**	**0.9509**	0.9746	**0.0662**	0.9954
512×512 Biased	**0.9661**	0.0789	**0.0273**	**0.9720**	0.0672	0.9506	**0.9759**	**0.0662**	**0.9966**

Table 21: Performance of PaGE ViT-H+ models trained with different heatmap resolutions.

M PaGE and Detection-based Methods

Tu et al. (2022) used a different formulation for gaze estimation and proposed HGTTR, a DETR-based joint head and gaze target detection model. Given an RGB image $x_{img} \in \mathbb{R}^{3\times H_{in}\times W_{in}}$ that contains at least one person, the model predicts a fixed number of N gaze instances. A gaze instance y consists of a predicted head bounding box $y_{bbox} \in [0,1]^4$, the probability that the object is indeed a head $y_{class} \in [0,1]$, an in/out prediction $y_{in/out} \in [0,1]$, and a gaze heatmap $y_H \in [0,1]^{H_{out}\times W_{out}}$. Similar formulations are also used by Tu et al. (2023) and Tonini et al. (2023). This task definition differs from the one we used (Section 3.1) in that the head bounding box is not part of the input, and is instead predicted by the model (along with its associated gaze heatmap).

M.1 Matching Instances at Train Time

To calculate loss at train time, HGTTR uses Hungarian algorithm to pair each predicted instance with a ground truth instance. The optimal matching for each ground truth instance y_i is the one that minimizes $\mathcal{L}_{match}(y_i, \hat{y}_{pred})$. In HGTTR, $\mathcal{L}_{match}$ is defined as:

$$\mathcal{L}_{match} = \lambda_1\mathcal{L}_{bbox} + \lambda_2\mathcal{L}_{class} + \lambda_3\mathcal{L}_{in/out} + \lambda_4\mathcal{L}_{heatmap}$$

where $\mathcal{L}_{bbox}$ is an IoU head box regression loss, $\mathcal{L}_{class}$ and $\mathcal{L}_{in/out}$ are binary classification losses, and $\mathcal{L}_{heatmap}$ is an L2 heatmap loss. Whatever the input, the model always predicts N instances. All methods in this line of work use $N = 20$.

M.2 Matching Instances at Test Time

The matching process at test time makes fair and direct comparison between detection-based methods and PaGE impossible. At test time, these detection-based methods use $\mathcal{L}_{match}$ to select the predicted instance for each ground truth instance, and evaluation metrics (AUC, L2, $AP_{in/out}$) are calculated accordingly. This process gives these methods a fundamental advantage over PaGE in that **the ground truth is used to select the optimal prediction that has the least loss for evaluation**. Ryan et al. (2025) observed that, since $N = 20$ is substantially larger than the usual number of ground truth instances in GazeFollow and VAT, Tonini et al. (2023)'s model often predicts *several* instances with the same head but different gaze targets. The matching process would then choose the instance with the correct prediction for metric calculation, giving the model an edge.

M.3 Performance Comparison

Ground Truth Gaze Matching	Model	GazeFollow AUC	GazeFollow Avg L2	GazeFollow Min L2	VideoAttentionTarget AUC	VideoAttentionTarget L2	VideoAttentionTarget AP
✓	Tu et al. (2022)	0.917	0.133	0.069	0.893	0.137	0.821
	Tu et al. (2023)	0.928	0.114	0.057	0.925	0.093	0.923
	Tonini et al. (2023)	0.922	0.069	0.029	0.933	0.104	0.934
×	Tonini et al. (2023)†	0.767	0.211	0.148	-	-	-
	PaGE (ours)	0.966	0.080	0.029	0.972	0.064	0.951

Table 22: Comparison between detection-based methods and PaGE. Since none of the detection-based methods were evaluated on ChildPlay, we omit that dataset in this comparison. †Evaluation done by Ryan et al. (2025) using Tonini et al. (2023)'s public code.

Despite using the conventional problem formulation and being systematically disadvantaged, PaGE still outperforms detection-based methods in most metrics, underlining the strong performance of

our method. When we remove ground-truth gaze matching, PaGE's advantage is even clearer. The results are in Table 22.