Spatio-Temporal Attention and Gaussian Processes for Personalized Video Gaze Estimation

Anonymous authors
Paper under double-blind review

Abstract

Gaze is an essential prompt for analyzing human behavior and attention. Recently, there has been an increasing interest in determining gaze direction from facial videos. However, video gaze estimation faces significant challenges, such as understanding the dynamic evolution of gaze in video sequences, dealing with static backgrounds, and adapting to variations in illumination. To address these challenges, we propose a simple and novel deep learning model designed to estimate gaze from videos, incorporating a specialized attention module. Our method employs a spatial attention mechanism that tracks spatial dynamics within videos. This technique enables accurate gaze direction prediction through a temporal sequence model, adeptly transforming spatial observations into temporal insights, thereby significantly improving gaze estimation accuracy. Additionally, our approach integrates Gaussian processes to include individual-specific traits, facilitating the personalization of our model with just a few labeled samples. Experimental results confirm the efficacy of the proposed approach, demonstrating its success in both within-dataset and cross-dataset settings. Specifically, our proposed approach achieves state-of-the-art performance on the Gaze360 dataset, improving by 2.5° without personalization. Further, by personalizing the model with just three samples, we achieved an additional improvement of 0.8°.

1 Introduction

The human gaze is an essential cue for conveying people's intent, making it promising for real-world applications such as human-robot interaction (Moon et al., 2014; Palinko et al., 2016), AR/VR (Patney et al., 2016; Padmanaban et al., 2017), and saliency detection (Rudoy et al., 2013; Parks et al., 2015). In addition, gaze plays a vital role in several computer vision tasks, including but not limited to object detection (Vasudevan et al., 2018), visual attention (Chong et al., 2018) and action recognition (Min & Corso, 2021). Despite the primary research emphasis on gaze estimation from images, the potential benefits of understanding the temporal dynamics of eye movements for video gaze estimation have been relatively overlooked. Constructing an accurate video-based gaze estimation model requires addressing the unique challenges inherent to videos. These include the evolution of eye movements throughout the video, correlations between gaze directions in successive frames, the predominance of a static background in most pixels, and variations due to individual-specific traits (Liu et al., 2018; Park et al., 2019b; Lindén et al., 2019). This work responds to these challenges by aiming to develop an accurate gaze estimation technique for videos using deep networks.

Realizing the potential of spatial and motion cues in videos, prior research has utilized residual frames and optical flows for several other vision tasks (Simonyan & Zisserman, 2014; Feichtenhofer et al., 2016; Wang et al., 2018). Specifically, these methods integrate RGB and residual frames as different input streams, requiring larger models with higher inference time and memory requirements (Karpathy et al., 2014; Wang et al., 2015; Girdhar et al., 2017). Similarly, 3D convolutional neural networks (CNNs) can also capture spatiotemporal information from videos, but they require many model parameters (Ji et al., 2012; Tran et al., 2015; Wang et al., 2017; Carreira & Zisserman, 2017; Feichtenhofer et al., 2019; Li et al., 2020). In addition, it is non-trivial to transfer knowledge from pre-trained 3D CNNs to new video tasks, as most pre-trained models rely on large 2D image datasets such as the ImageNet dataset (Deng et al., 2009). Despite

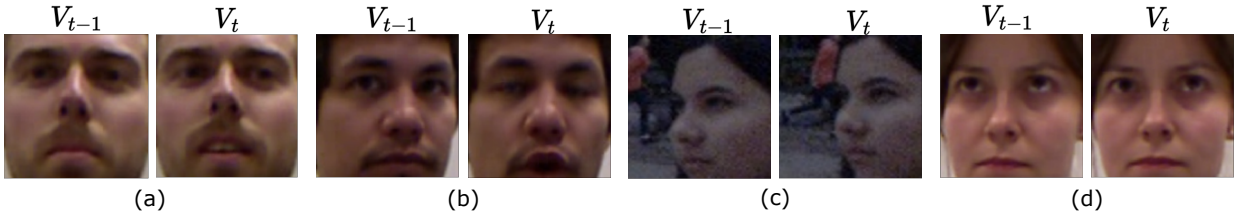


Figure 1: The figure illustrates a range of irrelevant factors for video gaze estimation, also referred to as distractors: (a) and (b) depict alterations in facial expression, (c) highlight background movement, and (d) represent a scenario without any distractors. These examples show the importance of accurately distinguishing between spatial changes due to eye movements and irrelevant distractors for video gaze estimation task.

the critical role of detecting spatial and motion cues in videos, there is a strong need to design efficient attention-based approaches for video-related tasks, including video gaze estimation.

In this work, we draw inspiration from the *change captioning* task to develop an approach for video gaze estimation. The change captioning task requires describing the changes between a pair of before and after images, expressed through a natural language sentence (Park et al., 2019a; Qiu et al., 2021; Tu et al., 2021). Both change captioning and gaze estimation tasks require differentiating irrelevant distractors, such as background movement and facial expression changes, from the relevant ones. Specifically, change captioning focuses on recognizing object movements, whereas gaze estimation concentrates on detecting eye movements. Similar to prior works (Park et al., 2019a; Qiu et al., 2021), our approach utilizes a spatial attention mechanism to focus on gaze-relevant information while minimizing the impact of distractors. For example, Figure 1 illustrates various distractors that may obfuscate gaze information in videos.

We introduce ***S**patio-**T**emporal **A**ttention for **G**aze **E**stimation (STAGE)*, a deep learning model for video gaze estimation. STAGE utilizes spatial changes in consecutive frames to integrate motion cues via a Spatial Attention Module (SAM) and captures global dynamics with a Temporal Sequence Model (TSM). The SAM module focuses on gaze-relevant information by applying local spatial attention between consecutive frames and effectively suppresses irrelevant distractors. Meanwhile, the TSM considers global dynamic movements across the temporal dimension, enabling enhanced prediction of gaze direction sequences. STAGE adeptly encodes motion information through the attention modules with fewer parameters than existing approaches like 3D CNNs (Ji et al., 2012) or two-branch networks (Karpathy et al., 2014), thus offering a more feasible solution for real-world applications.

To enhance the accuracy of gaze estimation models, previous studies have suggested personalization to address significant variability in individual-specific traits, such as eye geometry and appearance (Liu et al., 2018; Chen & Shi, 2019; Park et al., 2019b). Concretely, this is done by training a person-agnostic gaze model on a large labeled dataset and then fine-tuning it for individual users with a small set of labeled data. Consistent with this approach, we integrate Gaussian processes (GPs) (Rasmussen, 2004), known for their effectiveness in low-data scenarios, to personalize the STAGE model for individual users.

We use GPs to learn an additive bias correction and personalize the gaze estimate of the general STAGE model with just a few labeled samples. GPs enable the estimation of personalized 3D gaze directions and provide uncertainty measurements in interval form. These intervals represent a range of possible gaze directions instead of a single vector, making our approach more suitable for practical applications, such as monitoring attention on screens (Zhang et al., 2017b; Albiz et al., 2023). To evaluate the efficacy of the proposed STAGE model and personalization using GPs, we use three publicly available video gaze datasets: EYEDIAP (Funes Mora et al., 2014), Gaze360 (Kellnhofer et al., 2019) and EVE (Park et al., 2020).

In summary, our primary contributions are as follows:

- We introduce STAGE, a novel model for video gaze estimation. STAGE leverages an attention mechanism that is sensitive to spatial changes in sequential frames, effectively extracting gaze-relevant details from videos. This facilitates gaze prediction along the temporal axis for videos.

- We propose a sample-efficient approach to personalize the STAGE model, aiming to learn a bias correction model for gaze prediction using pre-trained Gaussian processes (Wang et al., 2021).

- Our approach either surpasses or matches to the state-of-the-art performance on three publicly available datasets for video gaze estimation. In particular, we obtain state-of-the-art results on the Gaze360 dataset in both cross-data and within-data experimental settings.

2 Related Work

Traditional methods of gaze estimation use an eye geometry model and exploit regression function to create a mapping from the eye or face images to the gaze vector (Guestrin & Eizenman, 2006; Hansen & Ji, 2009; Valenti et al., 2011; Nakazawa & Nitschke, 2012; Lu et al., 2016; Kar & Corcoran, 2017). While these methods perform well in controlled settings with consistent subject features, head positions, and lighting, their precision tends to drop in more varied and less controlled environments (Zhang et al., 2019).

Recently, with the emergence of deep learning methods, researchers employ CNNs to predict gaze direction from eye or face images directly (Tan et al., 2002; Zhang et al., 2015; Krafka et al., 2016; Huang et al., 2017; Fischer et al., 2018; Cheng et al., 2020). Image-based gaze estimation methods primarily use eye images to predict gaze directions (Zhang et al., 2015; Park et al., 2018a;b; Lian et al., 2018). Additionally, several approaches consider facial features such as head pose and facial appearance for estimating gaze information (Krafka et al., 2016; Zhang et al., 2017a; Ren et al., 2021; Gu et al., 2021). Generally, facial information for gaze estimation yields more accurate results than methods relying solely on eye images (Zhang et al., 2017a). Similarly, we also rely on full-face images for extracting gaze information in this work.

While gaze estimation techniques have shown impressive results with static images, the exploration of dynamic gaze estimation has been limited, primarily due to the scarcity of fully annotated gaze video datasets. Following the release of video gaze datasets (Funes Mora et al., 2014; Kellnhofer et al., 2019), several temporal gaze estimation models have emerged. These models are designed to predict the direction of eye gaze from a sequence of images. The initial work of Palmero et al. (2018) employs a recurrent CNN framework that concatenates the static features of each frame and feeds into a recurrent module, which is then used to predict the 3D gaze direction of the final frame in the sequence. Similarly, Kellnhofer et al. (2019) proposed a bidirectional LSTM that utilizes both past and future frames, indirectly incorporating spatial information.

Wang et al. (2019) released a dataset that captures human eye images and the corresponding ground-truth gaze positions on a screen while subjects engage in activities like browsing websites or watching videos. They proposed a dynamic gaze transition network to detect the transitions of eye movements over time and refine static gaze predictions using the dynamics learned from these transitions. Recently, Park et al. (2020) collected a large-scale video-based eye-tracking dataset with ground-truth Point of Gaze (PoG) on a screen and proposed a recurrent module to refine PoG estimates on video data. Our work aims to develop a deep learning approach for video gaze estimation by capturing the nuanced spatial and temporal dynamics.

As stated earlier, the performance of gaze estimators can be notably influenced by individual-specific traits, particularly when adapting these models to new subjects (Guestrin & Eizenman, 2006). However, in practical scenarios, there are typically only a few labeled samples available per subject and are insufficient for fine-tuning contemporary deep learning models, which tend to be over-parameterized (Park et al., 2019b). Previously, Liu et al. (2018) utilized a Siamese network to estimate gaze differences, employing a small number of calibration samples for personalization. Similarly, Park et al. (2019b) employed meta-learning techniques to achieve few-shot personalization, leveraging learned gaze embeddings. Chen & Shi (2019) introduced a method to model person-specific biases during the training phase, enabling personalization during testing with just a few samples. Our personalization approach is motivated by the efficacy of Gaussian processes in scenarios with limited data (Rasmussen, 2004). Unlike Chen & Shi (2019), our personalization approach outputs a different bias for each video frame and is designed to be compatible with any existing gaze estimation technique without necessitating alterations to the training objective.

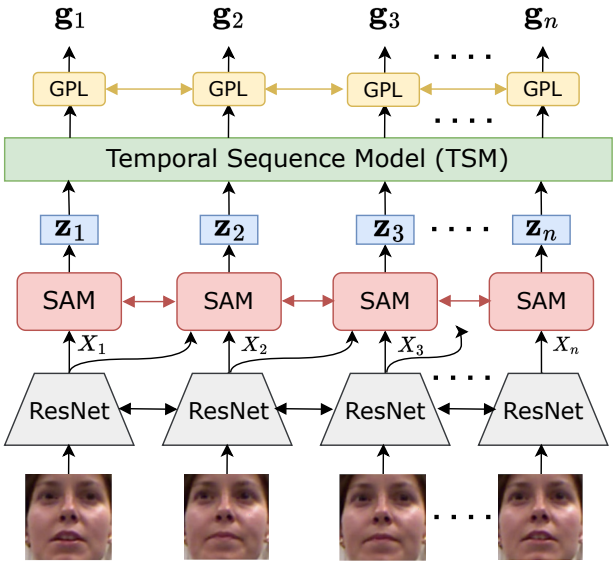


Figure 2: *A schematic overview of the proposed (person-agnostic) STAGE model.* The proposed model has three modules: spatial attention module (SAM), temporal sequence model (TSM), and gaze prediction layer (GPL). The SAM is designed to extract information relevant to the gaze by concentrating on the spatial differences between consecutive frames. In the figure, X_i represents features from ResNet, $\mathbf{z}_i$ denotes the motion-informed output of the SAM, and $\mathbf{g}_i$ corresponds to the predicted gaze direction.

3 Proposed Method

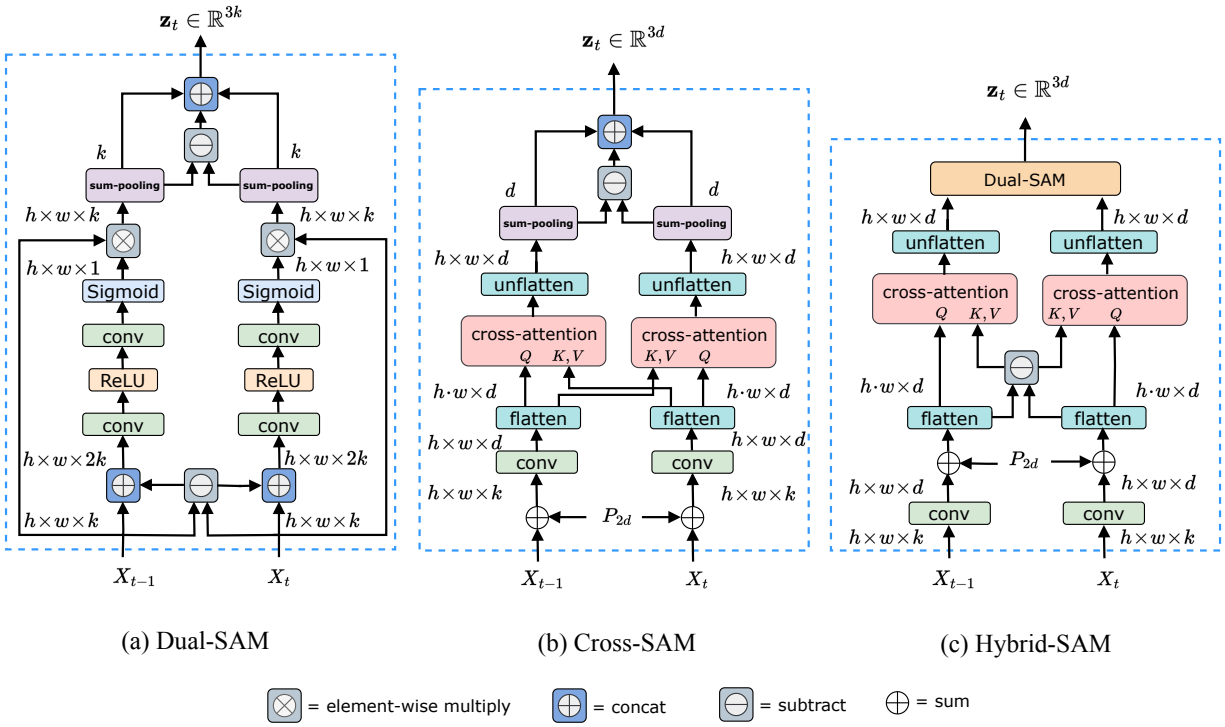
The main goal of video gaze estimation is to learn a deep network f defined as $f : V \mapsto G$ that maps a sequence of video frames $V \in \mathbb{R}^{n \times h_0 \times w_0 \times 3}$ to a sequence of gaze directions $G \in \mathbb{R}^{n \times 2}$, where n is the number of frames and h_0 and w_0 are height and width of each frame, respectively. The output gaze sequence G possesses pitch and yaw angles, which correspond to each frame in V.

The proposed STAGE model employs three modules for setting up the deep network f. Firstly, a ResNet-based CNN model receives the input video and extracts feature maps for all the frames. Then, in the following module of the STAGE model, we process feature maps using a *Spatial Attention Module* (SAM) to focus on the spatial motion information between consecutive frames followed by a *Temporal Sequence Model* (TSM) to learn temporal dynamics using past frame embeddings. Next, the gaze prediction layer (GPL) maps the features from the output of the TSM block to a sequence of gaze directions defined in terms of yaw and pitch angles. Figure 2 shows the schematic of the STAGE and its modules.

3.1 Spatial Attention Module (SAM)

Recall that SAM is aimed to distinguish gaze-relevant motion by analyzing differences between consecutive frames, focusing on crucial cues like eye or head movements for gaze estimation while filtering out irrelevant distractions like facial expressions or background movements. It aims to prioritize relevant video changes, particularly eye movements, and disregard non-essential ones.

First, we convert each frame in the video sequence V to features $X = [X_1, X_2, \ldots, X_n] \in \mathbb{R}^{n \times h \times w \times k}$, using the ResNet-based CNN model, where w, h, and k are the width, height, and the number of channels of the feature maps extracted by ResNet. The next step is to pass each consecutive feature pair (X_{t-1}, X_t) through a shared SAM. Concretely, the SAM module aggregates information from RGB features of X_{t-1} and X_t, and the feature differences $(X_t - X_{t-1})$ through a fusion strategy. Figure 3 provides an overview of all three SAM variants considered in this work. All SAM variants are optimized during model training and outputs $\mathbf{z}_t$, a feature representation with spatial motion information for the t^{th} frame of the video.



(a) Dual-SAM (b) Cross-SAM (c) Hybrid-SAM

Figure 3: **Block diagram of SAM variants.** For each variant, the input is a pair of the consecutive frame features X_{t-1} and X_t, and the output is a 1-dimensional feature vector encoding both RGB and motion information. P_{2d} are 2D positional embeddings with height and width that are the same as the input feature map. The *cross-attention* block in Cross-SAM and Hybrid-SAM is a standard transformer operation. The *sum-pooling* block applies feature pooling by summing them over height and width dimensions. In Hybrid-SAM, the keys and values for the cross-attention block are residual features, *i.e.*, the difference in features at t and $t-1$. This operation helps in capturing more nuanced spatial differences and motion cues.

Dual-SAM. Dual-SAM predicts separate spatial attention maps for both current X_t and past X_{t-1} frame. It compares the spatial attention maps of the current and past frames, and identifies the region that is most relevant to the observed motion changes. If the spatial attention maps are very similar, SAM infers that there is no substantial change between consecutive frames and encode these minimal differences in the output vector $\mathbf{z}_t \in \mathbb{R}^{3k}$. Conversely, if there is a difference, SAM incorporates this change into the output vector $\mathbf{z}_t$. This SAM variant is inspired by Park et al. (2019a) in the change captioning task and is shown in Figure-3a.

Cross-SAM. Unlike Dual-SAM, this variant utilizes cross-attention from transformer models (Vaswani et al., 2017) to encapsulate dense correlation between each pair of image patches in the past and current frames. This allows Cross-SAM to identify multiple changes between two frames, as opposed to Dual-SAM, which can only capture a single change. Practically, detecting multiple changes and subsequently filtering out irrelevant distractors is more useful for video gaze estimation tasks. Similar to the Dual-SAM, this variant utilizes both RGB and transformed motion signal at the output. Qiu et al. (2021) motivates the design of Cross-SAM and is shown in the Figure-3b.

Hybrid-SAM. The Hybrid-SAM combines the strengths of both Dual-SAM and Cross-SAM variants. Dual-SAM focuses on one local change, while Cross-SAM focuses on global context and captures multiple changes. Similar to Cross-SAM, Hybrid-SAM encapsulates multiple changes by applying a cross-attention mechanism using global context through position embeddings. However, unlike the Cross-SAM variant, it uses the difference between current and past frames as a key and value, emphasizing regions with the most significant motion differences. The Dual-SAM is utilized as a pooling operator to selectively focus on the most relevant changes, like eye or head movements, which are crucial for the task of gaze estimation.

The Hybrid-SAM is given in the Algorithm 1, and Dual-SAM and Cross-SAM are deferred to Appendix A.1. The input is features of the past frame X_{t-1} the and current frame X_t, respectively. Both input features are projected to the higher-dimensional feature maps using the convolution operation, and 2-D position embeddings $P_{2d} \in \mathbb{R}^{h \times w}$ are added (Line 1). Line 2 computes difference features X_{diff} for the video's t^{th} frame, and cross-attention is applied in Line 3. Lines 5-8 correspond to the same operations as in Dual-SAM.

Algorithm 1 Hybrid-Spatial Attention Module (Hybrid-SAM)

Input: X_{t-1}, X_t $\in \mathbb{R}^{h \times w \times k}$

Output: $\mathbf{z}_t$ $\in \mathbb{R}^{3 \cdot d}$

1: $X_{t-1} = \text{flat}(\text{conv}(X_{t-1}) + \mathbf{1}_{h,w} \odot P_{2d})$
$X_t = \text{flat}(\text{conv}(X_t) + \mathbf{1}_{h,w} \odot P_{2d})$ $\in \mathbb{R}^{h \cdot w \times d}$

2: $X_{\text{diff}} = X_t - X_{t-1}$

3: $X_{t-1} = \text{crossatten}(X_{t-1}, X_{\text{diff}}, X_{\text{diff}})$
$X_t = \text{crossatten}(X_t, X_{\text{diff}}, X_{\text{diff}})$ $\in \mathbb{R}^{h \cdot w \times d}$

4: $X_{t-1} = \text{unflat}(X_{t-1}, h \times w)$
$X_t = \text{unflat}(X_t, h \times w)$ $\in \mathbb{R}^{h \times w \times d}$

5: $X'_{t-1} = [X_{t-1}; X_t - X_{t-1}]$
$X'_t = [X_t; X_t - X_{t-1}]$ $\in \mathbb{R}^{h \times w \times 2 \cdot d}$

6: $A_{t-1} = \sigma(\text{conv}(\text{ReLU}(\text{conv}(X'_{t-1}))))$
$A_t = \sigma(\text{conv}(\text{ReLU}(\text{conv}(X'_t))))$ $\in \mathbb{R}^{h \times w \times 1}$

7: $\mathbf{v}_{t-1} = \sum_{h,w} A_{t-1} \odot X_{t-1}$
$\mathbf{v}_t = \sum_{h,w} A_t \odot X_t$ $\in \mathbb{R}^{d}$

8: $\mathbf{z}_t = [\mathbf{v}_{t-1}; \mathbf{v}_t - \mathbf{v}_{t-1}; \mathbf{v}_t]$ $\in \mathbb{R}^{3 \cdot d}$

9: **return** $\mathbf{z}_t$

σ denotes the sigmoid function, $\mathbf{1}_{h,w}$ is a one-hot vector spanning the spatial dimensions, $\odot$ is an element-wise dot product.

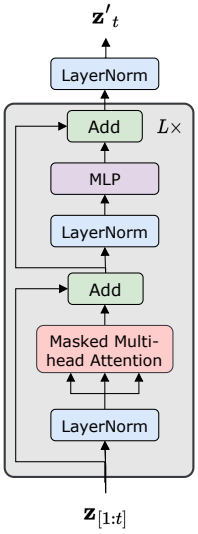


Figure 4: Block diagram of single transformer layer used in the temporal sequence model of the STAGE method. MLP is a Multi-Perceptron layer, and we use L blocks stacked together in the TSM.

3.2 Temporal Sequence Model (TSM)

The temporal sequence model subsumes spatially enhanced representations $\mathbf{z}_t$ produced by the SAM module and is intended to capture the temporal dynamics of the eye movements in the video. In particular, we consider two variants for TSM: recurrent neural networks (RNN) (Sutskever et al., 2014), and transformer network (Vaswani et al., 2017). The RNN consists of unidirectional LSTM layers (Hochreiter & Schmidhuber, 1997), and the transformer variant is a causal transformer decoder, which is prevalent in generative language modeling, such as the GPT-2 model (Radford et al., 2019).

We incorporate learned temporal position embeddings to enable the transformer model to discern temporal relationships within the input feature sequence. These embeddings are uniquely associated with each position, providing the model with explicit information about the relative ordering of elements within the sequence. The embedded features are then passed through multiple layers, each consisting of masked multi-head attention, LayerNorm (LN), and a Multi-Layer Perceptron (MLP) as shown in Figure 4. Masked multi-head attention allows the transformer model to attend to only past frame features. The output of the TSM is a feature sequence passed through an LN layer, similar to the GPT-2 model (Radford et al., 2019).

3.3 Gaze Prediction Layer and Training Objective

The gaze prediction layer is shared across all timestamps and uses an MLP to predict the gaze direction from the frame embeddings generated by the TSM module. For i^{th} sample and t^{th} frame, let $\{\mathbf{g}_t^i\}$ and $\{\hat{\mathbf{g}}_t^i\}$ denote the sequences of true and predicted gaze directions, respectively. Similarly, $\{\mathbf{p}_t^i\}$ and $\{\hat{\mathbf{p}}_t^i\}$ represent the sequences of true and predicted 2D Point-of-Gaze (PoG). We use the following objective function for

training STAGE model parameters (similar to Park et al., 2020):

$$\mathcal{L}_{final} = \frac{1}{b \cdot n} \sum_{i=1}^{b} \sum_{t=0}^{n-1} \frac{180}{\pi} \arccos\left(\frac{\mathbf{g}_t^{i^T} \hat{\mathbf{g}}_t^i}{|\mathbf{g}_t^i| \cdot |\hat{\mathbf{g}}_t^i|} \right) + \lambda \cdot ||\mathbf{p}_t^i - \hat{\mathbf{p}}_t^i|| \tag{1}$$

Here, λ is the weight parameter that controls the trade-off between 3D gaze angular error and 2D PoG mean absolute error. The second term is applied exclusively to datasets that have available ground-truth PoG.

3.4 Personalizing the STAGE model using Gaussian Processes

As previously mentioned, we propose person-specific Gaussian processes for modeling bias correction terms for each user, which operates on top of the proposed (person-agnostic) STAGE model. Specifically, if $f : V \mapsto G$ is the STAGE model, then the final prediction for person p is $\hat{\mathbf{f}}_p(V) = \mathbf{f}(V) + \mathbf{r}_p(V)$, where $\mathbf{r}_p$ is GP-based bias correction model for the person p, *i.e.*, it predicts the residual in addition to the model-agnostic prediction. The GP $\mathbf{r}_p$ models the components of gaze direction (*i.e.*, yaw and pitch) independently at the frame level, using two one-dimensional independent GPs. Concretely, $\mathbf{r}_p(V) = [(r_{p,\theta}(V_1), r_{p,\phi}(V_1)), (r_{p,\theta}(V_2), r_{p,\phi}(V_2)), \cdots, (r_{p,\theta}(V_n), r_{p,\phi}(V_n))]$, where $r_{p,\theta}$ and $r_{p,\phi}$ are the one-dimensional GP predictions for pitch and yaw components, respectively.

For GP hyper-parameter tuning and inference, we collect a set of training frames $\mathcal{D} = \{\mathbf{h}_i, y_i\}_{i=1}^{\ell}$ that are available for person p, where $\mathbf{h}_i \in \mathbb{R}^d$ are the flattened ResNet output features from the STAGE model, and y_i is either pitch or yaw of residual gaze angle, *i.e.*, $\mathbf{g}_i - \hat{\mathbf{g}}_i$, where, $\mathbf{g}_i$ and $\hat{\mathbf{g}}_i$ are true gaze direction and STAGE's predicted direction, respectively. To represent the dataset $\mathcal{D}$ in matrix format, we let $\mathbf{y} \in \mathbb{R}^\ell$ be the vector of residual angles, where the i^{th} entry equal to y_i, and $H \in \mathbb{R}^{\ell \times d}$ have its i^{th} row equal to the ResNet features $\mathbf{h}_i$. For brevity, we omit the person index p from henceforth discussion on GPs.

A Gaussian process associated with kernel (covariance) function $k(\mathbf{h}, \mathbf{h}') : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ is a distribution over functions that maps features to residual angles such that, for any $\mathbf{h}_1, \ldots, \mathbf{h}_\ell \in \mathbb{R}^d$:

$$\mathbf{r} = [r(\mathbf{h}_1), ..., r(\mathbf{h}_\ell)] \sim \mathcal{N}(\mu_0, K_H), \tag{2}$$

where $K_H = [k(\mathbf{h}_i, \mathbf{h}_j)]_{i,j=1}^{\ell} \in \mathbb{R}^{\ell \times \ell}$ is the kernel (covariance) matrix on the data points H, and r has a constant mean function with its value set to μ_0. The observed residual angle y_i is modeled as the i.i.d. Gaussian noise, *i.e.*, $y_i \sim \mathcal{N}(r(\mathbf{h}_i), \sigma^2 I)$. In particular, we use the (squared-exponential) automatic-relevance-determination (ARD) kernel, given as $k(\mathbf{h}, \mathbf{h}') = \tau \cdot e^{-\sum_{s=1}^{d} \frac{\left(\mathbf{h}(s) - \mathbf{h}'(s)\right)^2}{\theta(s)^2}}$, where τ and $\theta \in \mathbb{R}^d$ are kernel hyper-parameters. The ARD kernel's per-dimension scaling, being more expressive than the RBF kernel's use of a single length-scale, often leads to superior practical performance (Neal, 1998). Intuitively, this flexibility allows the model to adapt to varying feature relevance and noise levels, potentially leading to improved accuracy and generalization (Delbridge et al., 2019). Upon conditioning the GP model on the collected training dataset, the predictive posterior mean and covariance functions are as follows:

$$\textbf{mean: } \mu_{r|\mathcal{D}}(\mathbf{h}) = \mathbf{k}_\mathbf{h}^T (K_H + \sigma^2 I)^{-1} \mathbf{y}$$
$$\textbf{variance: } \sigma_{r|\mathcal{D}}(\mathbf{h}) = k(\mathbf{h}, \mathbf{h}) - \mathbf{k}_\mathbf{h}^T (K_H + \sigma^2 I)^{-1} \mathbf{k}_\mathbf{h}$$

where the vector $\mathbf{k}_\mathbf{h} \in \mathbb{R}^\ell$ has i^{th} entry $k(\mathbf{h}, \mathbf{h}_i)$, *i.e.*, kernel value between any feature vector $\mathbf{h}$ and i^{th} data point. The posterior mean function predicts the residual gaze angles and is utilized for correction. The posterior covariance function determines the uncertainty in this prediction, as illustrated in Figure 8.

Optimizing GP hyper-parameters using very few samples. GPs are non-parametric models and thus do not require tuning many parameters (Rasmussen, 2004). However, they still necessitate optimizing hyperparameters, which in our case are μ_0, σ, τ, and θ, totaling $d + 3$ hyperparameters as $|\theta| = d$. The ARD kernel adds flexibility to the GP model but also increases the number of hyperparameters to be tuned. Specifically, since $d = 16384$ when using features from the ResNet model, directly tuning hyperparameters using the log-likelihood of data $\mathcal{D}$ is prone to overfitting, particularly when as few as three samples are

present in $\mathcal{D}$. To overcome this challenge, we propose the application of pre-trained GPs, similar to the concurrent work by Wang et al. (2021). Pre-trained GPs entail the initial optimization of hyperparameters on data used for training the STAGE model, coupled with the implementation of early stopping during optimization to maximize the log-likelihood of dataset $\mathcal{D}$ for each individual. This methodology grants GPs flexibility with expressive ARD kernel and ensures a robust starting point due to pre-training.

4 Experiments and Results

4.1 Experimental Setup

Datasets. EVE (Park et al., 2020) is a large-scale video-based gaze dataset comprising over 12 million frames collected from 54 participants in a controlled indoor setting with four synchronized and calibrated camera views. Following the splits used by Park et al. (2020), there are 40 subjects in training and 6 subjects in the validation set. We discard the data from test subjects due to the unavailability of labels and evaluate our models on the validation set. Gaze360 (Kellnhofer et al., 2019) is a large-scale, physically unconstrained gaze dataset collected from 238 subjects in indoor and outdoor settings. The dataset includes a wide range of head poses, with 129K training images, 17K validation images, and 26K test images. We evaluate our models on all three subsets of the dataset: the full Gaze360 dataset, the front 180° subset, and the front 20° subset, as done by Kellnhofer et al. (2019). EyeDiap (Funes Mora et al., 2014) consists of 94 videos totaling 237 minutes, collected from 16 subjects in a laboratory environment. The EyeDiap dataset includes videos for both screen and floating targets and we select VGA videos of screen targets.

Implementation Details. The input video sequence V consists of 30 frames containing a full-face image of 128×128 pixels. We use ResNet-18 (Shafiq & Gu, 2022) initialized with GazeCLR (Jindal & Manduchi, 2023) weights shared between all timestamps to extract visual features from the image sequence. The third convolutional layer block of ResNet-18 outputs features with a dimension of $256 \times 8 \times 8$. We pass these features through the SAM module, followed by TSM and gaze prediction layers. We train STAGE end-to-end for 50K iterations using the SGD optimizer with an initial learning rate of 0.016 and momentum of 0.9. The learning rate is decayed using cosine annealing (Loshchilov & Hutter, 2017), and batch size is set to 16. We discuss more implementation details in Appendix A.2.

4.2 Evaluating the STAGE Model

In Section 4.2.1, we provide visual examples of attention maps superimposed on video frames, illustrating the qualitative impact of the SAM block in improving the overall performance of the STAGE. In addition to qualitative assessment, we provide quantitative evaluation of the SAM and TSM variants in two experimental settings: within-dataset (in Section 4.2.2) and cross-dataset (in Section 4.2.3). The primary objective of these experiments is to evaluate the effectiveness of incorporating a SAM block prior to the temporal sequence model in enhancing video gaze estimation accuracy. In Section 4.2.4, we also benchmark our proposed method against current leading methods in video gaze estimation for a within-dataset setting. Both qualitative and quantitative evaluation of GP-based personalization on EyeDiap participants is provided in Section 4.3. We discuss an ablation study on the number of SAM blocks in Appendix A.3.

Baselines. We benchmarked our framework against EyeNet (Park et al., 2020), which consists of ResNet-18 and RNN layers and uses both eye image patches as input. We adopted EyeNet to our setting and trained it on full-face images using $\mathcal{L}_{final}$ with $\lambda = 0.001$. We also train another variant of EyeNet by replacing the RNN module with a TSM similar to that used in our framework. For a fair comparison, we also implement EyeNet with our version of ResNet-18 initialized with GazeCLR (Jindal & Manduchi, 2023) weights and call it *EyeNet (GazeCLR)*. Further, we adapt the work of Chang et al. (2021) for gaze estimation, which introduces motion-aware-unit (MAU) for the video-prediction task. We also compare with a simple baseline by removing the SAM modules and concatenating X_t and $X_{\text{diff}} = (X_t - X_{t-1})$ before passing through TSM, termed *Concat-Residual*. Finally, we compare the three variants of SAM modules combined with two variants of TSM for cross-dataset and within-dataset experiments. For the sake of completion, we also evaluate the Hybrid-SAM method without the Dual-SAM module at the output, named as Hybrid-SAM†.

(a) Background person running (b) Face movement (c) Expression change (d) Expression change

Figure 5: Illustration of attention maps A_{t-1} and A_t, generated by the Hybrid-SAM, superimposed on sequential video frames V_{t-1} and V_t. The SAM module proficiently highlights the ocular area, key for analyzing eye movements, while simultaneously diminishing irrelevant distractions such as background motion (a), tongue movement (b), and changes in emotional expressions (c and d).

4.2.1 Qualitative Evaluation

We conducted a qualitative analysis primarily centered on assessing the Hybrid-SAM ability to distinguish between gaze-irrelevant distractors and gaze-relevant eye movements, which is crucial for video gaze estimation, as stated earlier. Specifically, we examined attention maps A_{t-1} and A_t, strategically overlaid on sequential video frames V_{t-1} and V_t, as depicted in Figure 5. We analyzed several frames showcasing scenarios from background activities to facial movements, all concurrent with dominant eye movements.

In Figure 5(a), the network adeptly focuses on eye movements in frame V_t (red pixels) and prior frame changes (blue pixels) despite significant background pixel shifts from a walking person. This underscores the effectiveness of spatial attention in filtering out irrelevant distractors to accurately identify subtle eye movements and gaze direction. As a result, it eases the process of temporal modeling in video gaze estimation. Additionally, as illustrated in Figure 5(b), although tongue movement presents a potential distraction, it is efficiently disregarded. Moreover, changes in facial expressions, depicted in Figure 5(c, d), are effectively overlooked by the Hybrid-SAM. These qualitative findings affirm that the spatio-temporal attention strategy adeptly minimizes significant distractions, particularly in the eye region, which is essential for accurately tracking gaze and eye movements in video gaze estimation tasks.

4.2.2 Within-dataset Evaluation

In the within-dataset experiments, we train and evaluate our model on the same domain dataset. Table 1 shows results for the within-dataset evaluation. We train our framework on the training subset of Gaze360 with $\lambda = 0$ and evaluate it over three test subsets as done in Kellnhofer et al. (2019). Our model demonstrates superior performance compared to the baseline models, including 'Concat-Residual', across all three subsets. Specifically, it achieves absolute improvements of 2.5°, 2.2° and 2.5° on full Gaze360, front 180° and front 20° subsets, respectively. Furthermore, it is noteworthy that Hybrid-SAM performs better in comparison to Hybrid-SAM†, illustrating the advantage of incorporating Dual-SAM as the pooling operator.

4.2.3 Cross-dataset Evaluation

We performed a cross-dataset evaluation, where the model was trained on the EVE dataset and evaluated on two different domain datasets, EyeDiap and Gaze360. Table 2 shows the comparison of mean angular errors (MAE) for the baselines and our proposed method. We observed a significant improvement in both datasets even with a simple concatenation of X_t and X_{diff}, *i.e.*, Concat-Residual approach outperforms EyeNet variants and MAU approach, which demonstrates that residual frames are an effective cue for video-gaze estimation.

The Dual-SAM and Cross-SAM variants show improvements over the Concat-Residual approach, indicating that the adapted methods are more accurate than naively using residual frames. Notably, the Hybrid-SAM approach improves over baselines by 1.2° in absolute and 14.28% in relative, on the EyeDiap dataset. It also outperformed the other Dual-SAM and Cross-SAM variants on all three evaluation sets. The last two columns of Table 2 show results on the full and front 180° Gaze360 subsets, which are similar to the subsets used in Kellnhofer et al. (2019). The Hybrid-SAM approach improved up to 3.6° on both subsets, further

Table 1: **Within-dataset Evaluation.** Comparison of mean angular errors (in degrees) between the proposed STAGE model, SAM and TSM variants, and other baseline approaches. Full, 180° and 20° are subsets of the Gaze360 dataset. Tx is the transformer TSM model. The **first** and second best results are bold-ed and underlined, respectively.

Method	**Full**	**180°**	**Front 20°**
EyeNet (Park et al., 2020)(GazeCLR)	12.53	12.08	9.45
EyeNet + Tx	13.00	12.55	9.73
Concat-Residual + LSTM	10.35	10.16	7.45
Concat-Residual + Tx	12.22	11.78	9.09
Dual-SAM + LSTM	10.12	9.92	7.08
Dual-SAM + Tx	10.13	9.93	7.23
Cross-SAM + LSTM	12.00	11.59	9.51
Cross-SAM + Tx	10.12	9.91	7.34
Hybrid-SAM† + LSTM	12.69	12.26	9.66
Hybrid-SAM† + Tx	12.33	11.90	9.53
Hybrid-SAM + LSTM	**10.05**	**9.84**	**6.92**
Hybrid-SAM + Tx	10.10	9.90	7.33

Table 2: **Cross-dataset Evaluation.** Comparison of mean angular gaze error (in degrees) between the proposed STAGE model, SAM and TSM variants, and other baseline approaches. Full and 180° are subsets of the Gaze360 dataset. Tx is the transformer TSM model. For each column, the **first** best result is bold-ed, and second best result is underlined.

Method	**EyeDiap**	**Full**	**180°**
MAU	21.30	34.18	33.57
EyeNet (Park et al., 2020)	16.07	31.37	30.77
EyeNet (GazeCLR)	7.74	26.57	25.95
EyeNet + Tx	8.40	26.25	25.64
Concat-Residual+ LSTM	7.12	24.12	23.52
Concat-Residual+ Tx	7.27	24.26	23.64
Dual-SAM + LSTM	7.04	24.18	23.58
Dual-SAM + Tx	6.77	23.99	23.38
Cross-SAM + LSTM	8.42	23.19	22.61
Cross-SAM + Tx	8.75	**22.57**	**22.01**
Hybrid-SAM† + LSTM	8.48	23.31	22.72
Hybrid-SAM† + Tx	7.79	22.66	22.09
Hybrid-SAM + LSTM	6.70	23.73	23.13
Hybrid-SAM + Tx	**6.54**	23.77	23.17

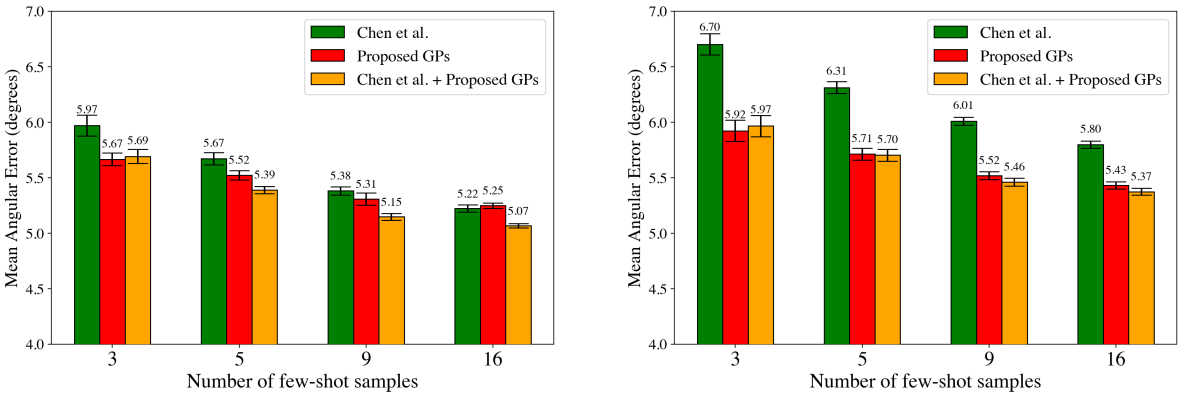
emphasizing the effectiveness of the SAM module. It is also worth noting that the performance improvements for the SAM variants hold for both LSTM and transformer-based TSM in both Tables 1 and 2. This shows that the SAM is helpful irrespective of the choice of the TSM model.

4.2.4 Comparison with State-of-the-art Video Gaze Estimation Methods

Table 3 compares the proposed STAGE method with state-of-the-art approaches for a within-dataset setting. Video-based gaze estimation methods such as the original work of Gaze360 (Kellnhofer et al., 2019) and MSA+Seq (Mishra & Lin, 2020) employ the LSTM model and learn through the Pinball loss function. We also compare our proposed gaze estimation approach with image-based methods such as L2CS-Net (Abdelrahman et al., 2022), both variants of GazeTR (Cheng & Lu, 2022), and self-supervised learning based method SwAT (Farkhondeh et al., 2022). We report the performance of these methods from the original

Table 3: **STAGE *vs.* State-of-the-art.** Comparison with state-of-the-art methods on Gaze360 data subsets under within-dataset setting (Tx = transformer-based TSM). The metric is the mean angular error (in degrees). The **first** and second best results are bold-ed and underlined, respectively.

Method	Full	180°	Front 20°
Gaze360 (Kellnhofer et al., 2019)	13.50	11.40	11.10
MSA+Seq (Mishra & Lin, 2020)	12.50	10.70	-
SwAT (Farkhondeh et al., 2022)	11.60	-	-
L2CS-Net (Abdelrahman et al., 2022)	-	10.41	9.02
GazeTR-Pure (Cheng & Lu, 2022)	-	13.58	-
GazeTR-Hybrid (Cheng & Lu, 2022)	-	10.62	-
Hybrid-SAM + LSTM	**10.05**	**9.84**	**6.92**
Hybrid-SAM + Tx	10.10	9.90	7.33



(a) Comparison of Dual-SAM + Tx

(b) Comparison of Hybrid-SAM + Tx

Figure 6: The figure shows the comparison of ℓ-shot GP personalization on the STAGE model with Chen & Shi (2020) for the EyeDiap dataset. The bars indicate the mean angular error (in degrees) and standard error over 10 iterations. The *Proposed GPs* consistently outperform the baseline for both SAM variants and achieve the best results when used in conjunction with Chen & Shi (2020).

work and show a comparison with our method. Our best results outperform these methods by 1.5°, 0.5° and 2.1° on full Gaze360, front 180° and front 20°, respectively. The superior performance of our method demonstrates the effectiveness of SAM and our choice for other components of the overall STAGE model.

4.3 Evaluating GPs for Personalization

As stated earlier, we first optimize the hyper-parameters of the GP model $\mathbf{r}_p$ for residual gaze direction prediction using the train subset of EVE dataset. Then, we adapt $\mathbf{r}_p$ for personalization on the EyeDiap participants. We randomly sample ℓ video frames for each participant 10 times and report the performance in Figure 6. We perform GP personalization on two SAM variants: Dual-SAM and Hybrid-SAM, using a transformer TSM model. The baseline method, proposed by Chen & Shi (2020), involves learning a single person-specific bias during training and utilizing a few labeled samples to predict bias during inference.

We obtain an absolute improvement of around 0.8° with the Hybrid-SAM over the baseline with as few as 3 samples. Applying GPs with the baseline objective, *i.e.*, "Chen *et al*+ GPs", we see consistent improvements over both GPs and the method proposed by Chen & Shi (2020). These results demonstrate that GPs' are a valuable tool and provide complementary strengths to Chen & Shi (2020). Unlike Chen & Shi (2020), GPs do not require altering the objective for training the deep network. They can be utilized for adaptation with any pre-trained existing model, such as STAGE.

For assessing the effectiveness of the GP model's uncertainty, we provide both qualitative and quantitative analysis of gaze predictions, as illustrated in Figures 7 and 8. Our evaluation begins with an analysis of the GP's posterior variance diagonal. We arrange this in ascending order and then apply different uncertainty thresholds to it. For each selected threshold, we compute the MAE on test samples that exhibit uncertainty levels below the threshold. This procedure is repeated across a range of different thresholds to evaluate performance. Figure 7 presents a comparison of the MAE for yaw and pitch against increasing fractions of test data samples. These samples are sorted according to the uncertainty in the GP prediction. This analysis demonstrates that GPs tend to deliver more accurate results when their variance is lower, signifying greater confidence in the predictions. Therefore, the uncertainty measure in the GP model can act as an effective indicator to avoid making inaccurate predictions.

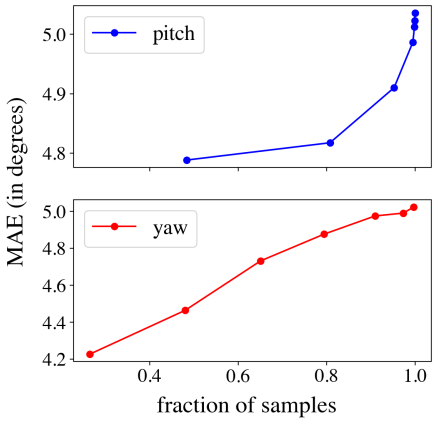
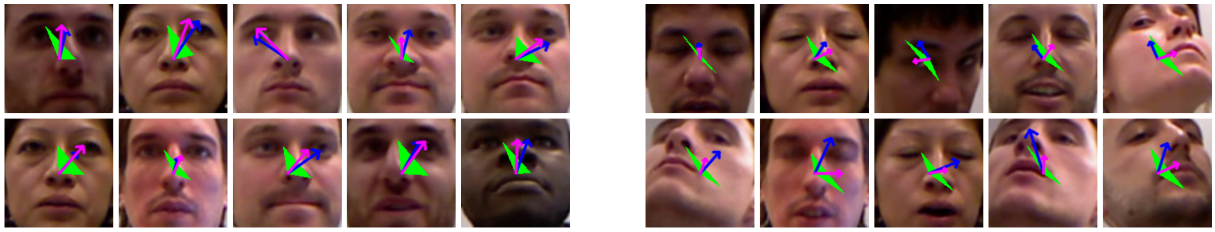


Figure 7: Comparison of Mean Angular Error (in degrees) of gaze components (yaw or pitch) with increasing fraction of test samples sorted with respect to the uncertainty of GP predictions. Plots exhibit that GPs are more accurate when the prediction is relatively more confident (with less variance).

We then examine the qualitative results depicted in Figure 8, which showcase the differences between confident and uncertain gaze predictions after personalization using the EyeDiap dataset. Notably, the uncertainty region typically includes the ground truth, as illustrated by the pink arrows falling within the green area. It is crucial to note that gaze predictions with higher uncertainty often align with situations that are challenging for human interpretation, like extreme head poses or closed eyes.



(a) Examples of certain predictions

(b) Examples of uncertain predictions

Figure 8: The figure depicts a few certain (a) and uncertain (b) predictions for gaze directions after GP's personalization on the EyeDiap dataset. The blue and pink arrows show ground truth and predicted gaze directions, respectively. The green-colored region offers uncertainty of the predictions in the pink arrows.

5 Conclusion

In this paper, we presented STAGE, a novel model for video gaze estimation, which utilizes an attention mechanism to encode spatial motion cues and temporal modelling. The method employed a spatial attention module to implicitly focus on the differences between consecutive frames, thereby highlighting relevant changes. We demonstrated that the performance of the STAGE model could be further enhanced using a few labeled samples with Gaussian processes. Future research could explore expanding the receptive field of the attention modules and integrating long-term spatial and temporal dynamics for further enhancements.

References

Ahmed A Abdelrahman, Thorsten Hempel, Aly Khalifa, and Ayoub Al-Hamadi. L2cs-net: Fine-grained gaze estimation in unconstrained environments. *arXiv preprint arXiv:2203.03339*, 2022.

Julius Albiz, Olga Viberg, and Andrii Matviienko. Guiding visual attention on 2d screens: Effects of gaze cues from avatars and humans. In *Proceedings of the 2023 ACM Symposium on Spatial User Interaction*, pp. 1–9, 2023.

Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6299–6308, 2017.

Zheng Chang, Xinfeng Zhang, Shanshe Wang, Siwei Ma, Yan Ye, Xiang Xinguang, and Wen Gao. Mau: A motion-aware unit for video prediction and beyond. *Advances in Neural Information Processing Systems*, 34:26950–26962, 2021.

Zhaokang Chen and Bertram Shi. Offset calibration for appearance-based gaze estimation via gaze decomposition. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 270–279, 2020.

Zhaokang Chen and Bertram E. Shi. Offset calibration for appearance-based gaze estimation via gaze decomposition. *2020 IEEE Winter Conference on Applications of Computer Vision*, pp. 259–268, 2019.

Yihua Cheng and Feng Lu. Gaze estimation using transformer. In *2022 26th International Conference on Pattern Recognition*, pp. 3341–3347. IEEE, 2022.

Yihua Cheng, Xucong Zhang, Feng Lu, and Yoichi Sato. Gaze estimation by exploring two-eye asymmetry. *IEEE Transactions on Image Processing*, 29:5259–5272, 2020.

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*, pp. 383–398, 2018.

Ian A. Delbridge, David S. Bindel, and Andrew Gordon Wilson. Randomly projected additive gaussian processes for regression. In *International Conference on Machine Learning*, 2019. URL `https://api.semanticscholar.org/CorpusID:209516346`.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Arya Farkhondeh, Cristina Palmero, Simone Scardapane, and Sergio Escalera. Towards self-supervised gaze estimation. In *British Machine Vision Conference*, 2022.

Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman. Convolutional two-stream network fusion for video action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1933–1941, 2016.

Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 6202–6211, 2019.

Tobias Fischer, Hyung Jin Chang, and Yiannis Demiris. Rt-gene: Real-time eye gaze estimation in natural environments. In *Proceedings of the European Conference on Computer Vision*, pp. 334–352, 2018.

Kenneth Alberto Funes Mora, Florent Monay, and Jean-Marc Odobez. Eyediap: A database for the development and evaluation of gaze estimation algorithms from rgb and rgb-d cameras. In *Proceedings of the symposium on eye tracking research and applications*, pp. 255–258, 2014.

Jacob R Gardner, Geoff Pleiss, David Bindel, Kilian Q Weinberger, and Andrew Gordon Wilson. Gpytorch: Blackbox matrix-matrix gaussian process inference with gpu acceleration. In *Advances in Neural Information Processing Systems*, 2018.

Rohit Girdhar, Deva Ramanan, Abhinav Gupta, Josef Sivic, and Bryan Russell. Actionvlad: Learning spatio-temporal aggregation for action classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 971–980, 2017.

Song Gu, Lihui Wang, Long He, Xianding He, and Jian Wang. Gaze estimation via a differential eyes' appearances network with a reference grid. *Engineering*, 7(6):777–786, 2021.

Elias Daniel Guestrin and Moshe Eizenman. General theory of remote gaze estimation using the pupil center and corneal reflections. *IEEE Transactions on biomedical engineering*, 53(6):1124–1133, 2006.

Dan Witzner Hansen and Qiang Ji. In the eye of the beholder: A survey of models for eyes and gaze. *IEEE transactions on pattern analysis and machine intelligence*, 32(3):478–500, 2009.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

Qiong Huang, Ashok Veeraraghavan, and Ashutosh Sabharwal. Tabletgaze: dataset and analysis for unconstrained appearance-based gaze estimation in mobile tablets. *Machine Vision and Applications*, 28: 445–461, 2017.

Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu. 3d convolutional neural networks for human action recognition. *IEEE transactions on pattern analysis and machine intelligence*, 35(1):221–231, 2012.

Swati Jindal and Roberto Manduchi. Contrastive representation learning for gaze estimation. In *Annual Conference on Neural Information Processing Systems*, pp. 37–49. PMLR, 2023.

Anuradha Kar and Peter Corcoran. A review and analysis of eye-gaze estimation systems, algorithms and performance evaluation methods in consumer platforms. *IEEE Access*, 5:16495–16519, 2017.

Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. Large-scale video classification with convolutional neural networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pp. 1725–1732, 2014.

Petr Kellnhofer, Adria Recasens, Simon Stent, Wojciech Matusik, and Antonio Torralba. Gaze360: Physically unconstrained gaze estimation in the wild. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6912–6921, 2019.

Kyle Krafka, Aditya Khosla, Petr Kellnhofer, Harini Kannan, Suchendra Bhandarkar, Wojciech Matusik, and Antonio Torralba. Eye tracking for everyone. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2176–2184, 2016.

Jun Li, Xianglong Liu, Mingyuan Zhang, and Deqing Wang. Spatio-temporal deformable 3d convnets with attention for action recognition. *Pattern Recognition*, 98:107037, 2020.

Dongze Lian, Lina Hu, Weixin Luo, Yanyu Xu, Lixin Duan, Jingyi Yu, and Shenghua Gao. Multiview multitask gaze estimation with deep convolutional neural networks. *IEEE transactions on neural networks and learning systems*, 30(10):3010–3023, 2018.

Erik Lindén, Jonas Sjostrand, and Alexandre Proutiere. Learning to personalize in appearance-based gaze tracking. In *Proceedings of the IEEE/CVF international conference on computer vision workshops*, pp. 0–0, 2019.

Gang Liu, Yuechen Yu, Kenneth Alberto Funes Mora, and Jean-Marc Odobez. A differential approach for gaze estimation with calibration. In *BMVC*, volume 2, pp. 6, 2018.

Ilya Loshchilov and Frank Hutter. SGDR: Stochastic gradient descent with warm restarts. In *International Conference on Learning Representations*, 2017.

Feng Lu, Yue Gao, and Xiaowu Chen. Estimating 3d gaze directions using unlabeled eye images via synthetic iris appearance fitting. *IEEE Transactions on Multimedia*, 18(9):1772–1782, 2016.

Kyle Min and Jason J Corso. Integrating human gaze into attention for egocentric activity recognition. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 1069–1078, 2021.

Ashesh Mishra and Hsuan-Tien Lin. 360-degree gaze estimation in the wild using multiple zoom scales. In *British Machine Vision Conference*, 2020.

AJung Moon, Daniel M Troniak, Brian Gleeson, Matthew KXJ Pan, Minhua Zheng, Benjamin A Blumer, Karon MacLean, and Elizabeth A Croft. Meet me where i'm gazing: how shared attention gaze affects human-robot handover timing. In *Proceedings of the 2014 ACM/IEEE international conference on Human-robot interaction*, pp. 334–341, 2014.

Atsushi Nakazawa and Christian Nitschke. Point of gaze estimation through corneal surface reflection in an active illumination environment. In *Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part II 12*, pp. 159–172. Springer, 2012.

Radford M. Neal. Assessing relevance determination methods using delve. 1998. URL `https://api.semanticscholar.org/CorpusID:59749732`.

Nitish Padmanaban, Robert Konrad, Tal Stramer, Emily A Cooper, and Gordon Wetzstein. Optimizing virtual reality for all users through gaze-contingent and adaptive focus displays. *Proceedings of the National Academy of Sciences*, 114(9):2183–2188, 2017.

Oskar Palinko, Francesco Rea, Giulio Sandini, and Alessandra Sciutti. Robot reading human gaze: Why eye tracking is better than head tracking for human-robot collaboration. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5048–5054. IEEE, 2016.

Cristina Palmero, Javier Selva, Mohammad Ali Bagheri, and Sergio Escalera. Recurrent cnn for 3d gaze estimation using appearance and shape cues. In *British Machine Vision Conference*, 2018.

Dong Huk Park, Trevor Darrell, and Anna Rohrbach. Robust change captioning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4624–4633, 2019a.

Seonwook Park, Adrian Spurr, and Otmar Hilliges. Deep pictorial gaze estimation. volume 11217 lncs. 2018a.

Seonwook Park, Xucong Zhang, Andreas Bulling, and Otmar Hilliges. Learning to find eye region landmarks for remote gaze estimation in unconstrained settings. In *Proceedings of the 2018 ACM symposium on eye tracking research & applications*, pp. 1–10, 2018b.

Seonwook Park, Shalini De Mello, Pavlo Molchanov, Umar Iqbal, Otmar Hilliges, and Jan Kautz. Few-shot adaptive gaze estimation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 9368–9377, 2019b.

Seonwook Park, Emre Aksan, Xucong Zhang, and Otmar Hilliges. Towards end-to-end video-based eye-tracking. In *Computer Vision–ECCV 2020: 16th European Conference, Proceedings, Part 16*, pp. 747–763. Springer, 2020.

Daniel Parks, Ali Borji, and Laurent Itti. Augmented saliency model using automatic 3d head pose detection and learned gaze following in natural scenes. *Vision research*, 116:113–126, 2015.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.

Anjul Patney, Marco Salvi, Joohwan Kim, Anton Kaplanyan, Chris Wyman, Nir Benty, David Luebke, and Aaron Lefohn. Towards foveated rendering for gaze-tracked virtual reality. *ACM Transactions on Graphics (TOG)*, 35(6):1–12, 2016.

Yue Qiu, Shintaro Yamamoto, Kodai Nakashima, Ryota Suzuki, Kenji Iwata, Hirokatsu Kataoka, and Yutaka Satoh. Describing and localizing multiple changes with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1971–1980, 2021.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

Carl Edward Rasmussen. *Gaussian Processes in Machine Learning*, pp. 63–71. Springer Berlin Heidelberg, Berlin, Heidelberg, 2004.

Dakai Ren, Jiazhong Chen, Jian Zhong, Zhaoming Lu, Tao Jia, and Zongyi Li. Gaze estimation via bilinear pooling-based attention networks. *Journal of Visual Communication and Image Representation*, 81:103369, 2021.

Dmitry Rudoy, Dan B Goldman, Eli Shechtman, and Lihi Zelnik-Manor. Learning video saliency from human gaze using candidate selection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1147–1154, 2013.

Muhammad Shafiq and Zhaoquan Gu. Deep residual learning for image recognition: a survey. *Applied Sciences*, 12(18):8972, 2022.

Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. *Advances in neural information processing systems*, 27, 2014.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. *Advances in neural information processing systems*, 27, 2014.

Kar-Han Tan, David J Kriegman, and Narendra Ahuja. Appearance-based eye gaze estimation. In *Sixth IEEE Workshop on Applications of Computer Vision, 2002. Proceedings.*, pp. 191–195. IEEE, 2002.

Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pp. 4489–4497, 2015.

Yunbin Tu, Tingting Yao, Liang Li, Jiedong Lou, Shengxiang Gao, Zhengtao Yu, and Chenggang Yan. Semantic relation-aware difference representation learning for change captioning. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pp. 63–73, 2021.

Roberto Valenti, Nicu Sebe, and Theo Gevers. Combining head pose and eye location information for gaze estimation. *IEEE Transactions on Image Processing*, 21(2):802–815, 2011.

Arun Balajee Vasudevan, Dengxin Dai, and Luc Van Gool. Object referring in videos with language and human gaze. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4129–4138, 2018.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Kang Wang, Hui Su, and Qiang Ji. Neuro-inspired eye tracking with eye movement dynamics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9831–9840, 2019.

Limin Wang, Yu Qiao, and Xiaoou Tang. Action recognition with trajectory-pooled deep-convolutional descriptors. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4305–4314, 2015.

Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks for action recognition in videos. *IEEE transactions on pattern analysis and machine intelligence*, 41(11):2740–2755, 2018.

Xuanhan Wang, Lianli Gao, Peng Wang, Xiaoshuai Sun, and Xianglong Liu. Two-stream 3-d convnet fusion for action recognition in videos with arbitrary size and length. *IEEE Transactions on Multimedia*, 20(3): 634–644, 2017.

Zi Wang, George E Dahl, Kevin Swersky, Chansoo Lee, Zelda Mariet, Zachary Nado, Justin Gilmer, Jasper Snoek, and Zoubin Ghahramani. Pre-trained gaussian processes for bayesian optimization. *arXiv preprint arXiv:2109.08215*, 2021.

Yuxin Wu and Kaiming He. Group normalization. In *Proceedings of the European conference on computer vision*, pp. 3–19, 2018.

Xucong Zhang, Yusuke Sugano, Mario Fritz, and Andreas Bulling. Appearance-based gaze estimation in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4511–4520, 2015.

Xucong Zhang, Yusuke Sugano, Mario Fritz, and Andreas Bulling. It's written all over your face: Full-face appearance-based gaze estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 51–60, 2017a.

Xucong Zhang, Yusuke Sugano, and Andreas Bulling. Evaluation of appearance-based methods and implications for gaze-based applications. In *Proceedings of the 2019 CHI conference on human factors in computing systems*, pp. 1–13, 2019.

Yanxia Zhang, Ken Pfeuffer, Ming Ki Chong, Jason Alexander, Andreas Bulling, and Hans Gellersen. Look together: using gaze for assisting co-located collaborative search. *Personal and Ubiquitous Computing*, 21: 173–186, 2017b.

A Appendix

A.1 Proposed Method – Omitted Details

We provide the mathematical formulation of Dual-SAM and Cross-SAM in Algorithm 2 and 3, respectively.

Algorithm 2 Dual-Spatial Attention Module (Dual-SAM)

Input: X_{t-1}, X_t $\in \mathbb{R}^{h\times w\times k}$

Output: $\mathbf{z}_t$ $\in \mathbb{R}^{3\cdot k}$

1: $X'_{t-1} = [X_{t-1}; X_t - X_{t-1}]$

$X'_t \;\; = [X_t; X_t - X_{t-1}]$ $\in \mathbb{R}^{h\times w\times 2\cdot k}$

2: $A_{t-1} = \sigma(\text{conv}(\text{ReLU}(\text{conv}(X'_{t-1}))))$

$A_t = \sigma(\text{conv}(\text{ReLU}(\text{conv}(X'_t))))$ $\in \mathbb{R}^{h\times w\times 1}$

3: $\mathbf{v}_{t-1} = \sum_{h,w} A_{t-1} \odot X_{t-1}$

$\mathbf{v}_t = \sum_{h,w} A_t \odot X_t$ $\in \mathbb{R}^{k}$

4: $\mathbf{z}_t = [\mathbf{v}_{t-1}; \mathbf{v}_t - \mathbf{v}_{t-1}; \mathbf{v}_t]$ $\in \mathbb{R}^{3\cdot k}$

5: **return** $\mathbf{z}_t$

A.2 Additional Implementation Details

The Dual-SAM consists of two convolutional layers with kernel size 1 and output feature maps of 64 and 1, respectively. The first convolutional layer has a group normalization layer (Wu & He, 2018) applied to the output features, followed by a dropout layer with $p = 0.5$. In Cross-SAM and Hybrid-SAM, we project the

Algorithm 3 Cross-Spatial Attention Module (Cross-SAM)

Input: X_{t-1}, X_t $\in \mathbb{R}^{h\times w\times k}$
Output: $\mathbf{z}_t$ $\in \mathbb{R}^{3\cdot d}$

$X_{t-1} = \text{flat}(\text{conv}(X_{t-1}) + 1_{h,w} \odot P_{2d})$
$X_t \quad = \text{flat}(\text{conv}(X_t) + 1_{h,w} \odot P_{2d})$ $\in \mathbb{R}^{h\cdot w\times d}$
$X_{t-1} = \text{crossatten}(X_{t-1}, X_t, X_t)$
$X_t \quad = \text{crossatten}(X_t, X_{t-1}, X_{t-1})$ $\in \mathbb{R}^{h\cdot w\times d}$
$\mathbf{v}_{t-1} = \sum_{h,w} \text{unflat}(X_{t-1}, h \times w)$
$\mathbf{v}_t = \sum_{h,w} \text{unflat}(X_t, h \times w)$ $\in \mathbb{R}^{d}$
$\mathbf{z}_t = [\mathbf{v}_{t-1}; \mathbf{v}_t - \mathbf{v}_{t-1}; \mathbf{v}_t]$ $\in \mathbb{R}^{3\cdot d}$
return $\mathbf{z}_t$

incoming features to higher channels through a convolution layer with $d = 512$ and a kernel size 1. After adding 2D positional embeddings to the projected feature maps, they go through the cross-attention encoder, which consists of four heads and two layers with an embedding size of 64.

The TSM model has two variants: an LSTM variant and a transformer variant. The LSTM variant consists of one unidirectional LSTM layer with a hidden dimension of 128. The transformer variant is based on GPT-2 (Radford et al., 2019) network with 6-heads and 6-layers, operating on a dimension of $d = 128$, and initialized randomly. The gaze prediction layer consists of two fully connected (FC) layers. The first FC layer has a SeLU activation function and a hidden dimension of the same size as the input dimension. The second FC layer outputs the 2D gaze direction angles, pitch and yaw.

Our STAGE model is implemented in PyTorch (Paszke et al., 2017). We set $\lambda = 0.001$ for cross-data and $\lambda = 0$ for within-data evaluations. For GP hyper-parameter optimization, we use Adam optimizer with a learning rate of 0.001, implemented using GPytorch (Gardner et al., 2018). Our code and trained models will be made publicly available in the future and are zipped in supplementary.

A.3 Ablation Study

In the ablation study, we study the impact of adding multiple SAM blocks in the STAGE model, where the output of one SAM goes as input to the next. The ablation study on the number of Dual- and Hybrid-SAM blocks (four blocks vs. one block) for within-data and cross-data settings are shown in Tables 4(a) and (b), respectively. We observe no significant improvements over a single block of SAM, indicating that one SAM block is enough to provide spatial motion cues between consecutive frame features and improve performance.

Table 4: **Ablation Study:** Comparison of different numbers of SAM blocks employed in our STAGE method. Tx is transformer-based TSM, and training is performed for within-data and cross-data settings in (a) and (b), respectively. The metric reported is mean angular errors (in degrees).

(a) Within-dataset evaluation

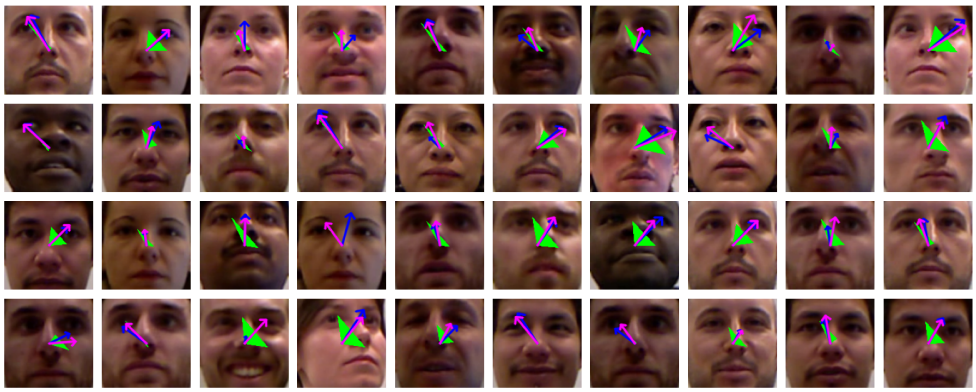
Method	Full	180°	20°
Dual-SAM(1-block)+Tx	10.13	9.93	7.23
Hybrid-SAM(1-block)+Tx	10.10	9.90	7.33
Dual-SAM(4-blocks)+Tx	12.13	11.68	9.33
Hybrid-SAM(4-blocks)+Tx	10.25	10.08	7.27

(b) Cross-dataset evaluation

Method	EyeDiap	Full	180°
Dual-SAM(1-blocks)+Tx	6.77	23.99	23.38
Hybrid-SAM(1-blocks)+Tx	6.54	23.77	23.17
Dual-SAM(4-blocks)+Tx	7.27	23.34	22.74
Hybrid-SAM(4-blocks)+Tx	7.55	23.52	22.91

A.4 More Visualizations

Here, we provide additional visualizations of the predictions from personalized GP on top of the STAGE model, similar to Figure 8 in the main manuscript. Figure 9a and 9b respectively show certain and uncertain prediction images from the EYEDIAP dataset after performing GP personalization. The ground truth and



(a) Certain predictions for EYEDIAP dataset

(b) Uncertain Predictions for EYEDIAP dataset

Figure 9: The figure depicts a few confident 9a and uncertain 9b predictions for gaze directions after GP's personalization on the EYEDIAP dataset. Blue and pink arrows show ground truth and predicted gaze directions, respectively. The green-colored region offers uncertainty of the predictions in pink arrows. The uncertainty region often covers the ground truth, *i.e.*, the pink arrows are in the green-colored area.

predicted gaze directions are respectively shown with blue and pink colored arrows, and the corresponding uncertainty of prediction is shown with the green colored triangle.