
DiffEye: Diffusion-Based Continuous Eye-Tracking Data Generation Conditioned on Natural Images

Ozgur Kara* Harris Nisar* James M. Rehg
{ozgurk2, nisar2, jrehg}@illinois.edu
University of Illinois Urbana-Champaign

Abstract

Numerous models have been developed for scanpath and saliency prediction, which are typically trained on scanpaths, which model eye movement as a sequence of discrete fixation points connected by saccades, while the rich information contained in the raw trajectories is often discarded. Moreover, most existing approaches fail to capture the variability observed among human subjects viewing the same image. They generally predict a single scanpath of fixed, pre-defined length, which conflicts with the inherent diversity and stochastic nature of real-world visual attention. To address these challenges, we propose DiffEye, a diffusion-based training framework designed to model continuous and diverse eye movement trajectories during free viewing of natural images. Our method builds on a diffusion model conditioned on visual stimuli and introduces a novel component, namely *Corresponding Positional Embedding (CPE)*, which aligns spatial gaze information with the patch-based semantic features of the visual input. By leveraging raw eye-tracking trajectories rather than relying on scanpaths, DiffEye captures the inherent variability in human gaze behavior and generates high-quality, realistic eye movement patterns, despite being trained on a comparatively small dataset. The generated trajectories can also be converted into scanpaths and saliency maps, resulting in outputs that more accurately reflect the distribution of human visual attention. DiffEye is the first method to tackle this task on natural images using a diffusion model while fully leveraging the richness of raw eye-tracking data. Our extensive evaluation shows that DiffEye not only achieves state-of-the-art performance in scanpath generation but also enables, for the first time, the generation of continuous eye movement trajectories. Project webpage: <https://diff-eye.github.io/>

1 Introduction

Humans obtain visual information via a sequence of eye movements that direct the fovea to analyze important elements of the visual scene. The resulting pattern of fixations and saccades¹ is dependent upon the task at hand (e.g., free exploration, visual search, etc.) and determines the visual elements that receive our attention [1, 2]. The gold standard for quantifying visual attention is eye-tracking, which records a subject’s eye movements as they view an image (stimulus) on a monitor. Eye tracking produces a trajectory of eye movements, consisting of a sequence of (x, y) pairs, which is processed to identify the fixations and saccades. Eye tracking provides valuable insights into the properties of the visual scene that drive human attention, and is widely-used for applications in virtual reality [3, 4, 5], foveated rendering [6, 7], and advertising [8, 9]. Eye tracking is also widely used in psychology to identify differences in how visual information is processed by individuals [10]. For example, eye tracking has been used to study social cognition in conditions such as autism by identifying differences in how social cues are processed [11, 12] (e.g., individuals with autism have been shown

*Equal contribution

¹Fixations occur when the eye remains relatively still and attention is focused on a specific location. Saccades are rapid eye movements between fixations that enable quick shifts of focus between regions of interest.

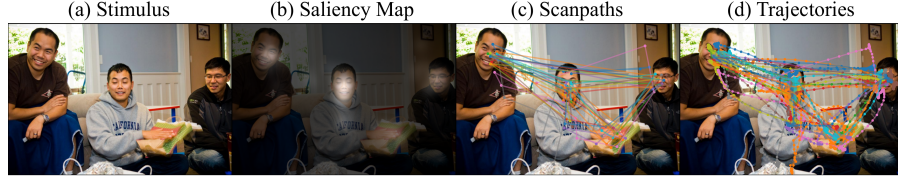


Figure 1: **Comparison of different eye-tracking data types.** (a) Original visual stimulus. (b) Saliency maps highlight regions of interest but do not capture the temporal dynamics of human attention. (c) Scanpaths offer a compressed representation of eye movement trajectories. (d) Full eye movement trajectories, recorded via eye trackers, provide detailed insights into attention dynamics. This example is from the MIT1003 dataset [18]; each color represents a different subject, emphasizing the importance of modeling inter-subject variability.

to exhibit atypical patterns of attention to faces [13, 14]). Although eye-tracking studies provide high-quality data, they are costly and time-consuming to conduct, and require significant expertise for proper experimental design [10]. As a result, there has been growing interest in developing computational models that can predict visual attention to images.

Modeling human visual attention has traditionally focused on saliency prediction [15, 16]. In this task, given an image, the goal is to produce a heat map, known as a saliency map, where high values correspond to regions that are more likely to receive fixations. However, these spatial summaries neglect the temporal structure of eye movements, which is critical for applications in virtual reality and psychology research. For example, recent work in developmental science has shown that children with and without autism demonstrate different spatio-temporal gaze behaviors [17]. To capture the temporal dimension, researchers have investigated the scanpath prediction task, which involves predicting a sequence of fixations for a given image. Scan path prediction has historically involved generating a single deterministic output, which is insufficient to capture the inherent variability of human attention. Figure 1 illustrates the variation in both scanpaths (c) and eye movement trajectories (d) for a population of viewers, with each color representing a different subject. As a result, recent work has shifted towards generative modeling to better reflect the stochastic nature of eye movements. However, existing approaches to the generative modeling of visual attention suffer from two major limitations: First, most prior methods operate directly on scanpath data rather than on eye movement trajectories, resulting in the loss of valuable information for spatio-temporal modeling. For example, in the MIT1003 dataset [18], the average scanpath consists of 8.4 ± 2.6 timesteps, while the raw trajectories average 723.7 ± 13.4 timesteps, indicating a substantial reduction in spatio-temporal information. Second, many existing methods model scanpath variability either autoregressively by sampling intermediate outputs, such as saliency maps derived from fixation history [19, 20, 21], or perform deterministic scanpath prediction [22], both of which fall short in accurately capturing the true distribution of gaze behavior.

In this paper, we hypothesize that generative modeling using the full eye movement trajectories produced by eye tracking can yield more effective representations of visual attention dynamics, in comparison to prior works that operate on scanpath data. In particular, we demonstrate that *training on eye movement trajectories results in increased accuracy in scanpath prediction*. In addition, we argue that generative models are better suited than discriminative models for learning from this rich temporal information, particularly given the variability present in both visual tasks and data collection processes. To evaluate these hypotheses, we introduce *DiffEye*, a diffusion-based framework that is conditioned on a given visual stimulus and trained directly on raw eye movement trajectories to generate realistic eye-tracking samples. These generated trajectories can then be transformed into scanpaths or saliency maps. To enhance stimulus conditioning and improve the interaction between trajectories and the image via cross-attention, we propose a novel positional embedding strategy called *Corresponding Positional Embedding (CPE)*. This method strengthens conditioning by aligning gaze information with the semantic features of the stimulus. In addition, we use high resolution image features to improve semantic precision, leading to more accurate trajectory generation.

In summary, our key contributions are as follows:

- We present the first diffusion-based training framework that directly utilizes the raw eye movement trajectories obtained from standard eye tracking datasets. Our method outper-

forms existing approaches in scanpath generation on unseen datasets, verifying its generalizability, and enables the synthesis of continuous eye-tracking data for natural images.

- We introduce a novel positional embedding strategy, *Corresponding Positional Embedding (CPE)*, which improves conditioning on visual stimuli by aligning spatial locations in both the image and trajectory space. Notably, our model achieves strong results even when trained on relatively small datasets.
- Unlike recent models that rely on autoregressive sampling to capture scanpath variability or produce a single deterministic prediction, our approach leverages diffusion to model the inherent variability in human gaze behavior across subjects. As a result, it can generate eye movement trajectories that can be converted into scanpaths of variable lengths, producing diverse and plausible patterns that reflect the natural variability in human attention.

2 Related Works

Eye Movement Trajectory Generation The closest related work to this paper is DiffGaze [23], which adopts a diffusion-based approach to generating eye movement trajectories in 360° images. While the focus on 360° images is innovative, the resulting models cannot be applied to the analysis of the standard RGB images used in eye tracking research. Moreover, the conditioning approach in DiffGaze relies on a single global feature vector obtained via spherical convolution. In contrast, our approach enhances conditioning through the use of CPE and patch level image features. Our ablation studies (see Table 2) demonstrate the importance of these elements for achieving strong performance. Unfortunately, a direct comparison to DiffGaze is not possible, since the code and pretrained models are not publicly available. We hope that the release of our code and models, along with training and testing splits for eye movement trajectory data from standard datasets, will spur additional research on this understudied problem.

Scanpath Modeling Works on scanpath generation operate directly on the sequences of fixations that are extracted by postprocessing eye-tracking data [24]. Given an input image, these models predict the location and sequence of fixations comprising the generated scanpath. Previous works have explored a variety of methods and problem formulations. IOR-ROI [25] uses LSTMs to predict scanpaths but outputs scanpaths of fixed length which limits its generalizability. DeepGaze III [19] samples from saliency maps using the fixation history, and as a result it cannot utilize the global image representation in predicting each fixation. Gazeformer [22] introduces ZeroGaze for unseen targets, but their deterministic method cannot produce a distribution of trajectories for a given stimulus. Some works on scanpath generation have addressed the fact that gaze behavior is dependent upon the task. Similar to us, PathGAN [26] uses GANs [27] for generative modeling in the FV case, though training is unstable. At present, the only available datasets containing eye movement trajectories address the standard free viewing (FV) task. Therefore, an investigation of the performance of our DiffEye approach on non-FV tasks will depend on future data collection efforts. In contrast, Chen et al. [21] use reinforcement learning for target present (TP) and visual QA tasks. Other methods [28, 29] support TP and the related target absent (TA) condition, but not FV. HAT [20] is unique in covering the three tasks FV, TP, and TA with a single model via a task query, but follows a similar sampling strategy as DeepGaze III to generate scanpaths. In summary, no prior scanpath generation work has taken advantage of the benefits of diffusion models, and competing methods for the FV task have a variety of limitations. Our experiments (see Table 3) compare scanpaths extracted from DiffEye to prior scanpath generation methods and demonstrate superior performance across a variety of metrics. We also note that our approach of directly generating eye movement trajectories gives end users the flexibility to define and extract fixations using any desired approach.

There have been a number of prior scanpath generation works that target 360° images. Similar to the discussion of DiffGaze above, these methods are not directly comparable to ours and are included here for the sake of completeness. SaltiNet [30] samples from saliency volumes; Pathformer3D [31] uses transformers for autoregressive prediction. ScanGAN360 [32] and Scantd [33] apply GANs and diffusion to the 360° modeling task. Finally, we note that most prior works including ours have used aggregate eye tracking data across multiple users to model attention at the population level. In contrast, a recent work from Chen et. al. [34] develops an approach to scanpath prediction for a single user. The specialization of DiffEye to the 360° image and individual prediction tasks remain as interesting subjects for future research.

Saliency Modeling Prior works on saliency prediction have benefited from a deep connection to research in visual perception [35, 36, 37, 1, 38]. Itti’s seminal work [39] later drew interest from the computer vision community [40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50]. As saliency methods capture only spatial attention and ignore the temporal dynamics central to eye movements, they are not able to address the needs of the diverse applications that motivate this work. Saliency maps can be derived post hoc from the output of our DiffEye method via fixation extraction [51] and spatial aggregation. In this regard, DiffEye can be viewed as an alternative approach to saliency prediction based on the detailed modeling of gaze behavior. However, methods that are designed to directly optimize saliency prediction performance are capable of higher accuracy than an indirect approach based on fixations. Further, the saliency baselines we compared against [47, 48, 49, 50] were trained on combinations of various saliency datasets [18, 43, 52, 53, 54, 55], including MIT1003, while our method is only trained on the relatively smaller MIT1003 dataset.

3 Methods

3.1 Preliminaries

Given a dataset $\mathcal{D} = \{(I^i, \{R_j^i\}_{j=1}^{K^i})\}_{i=1}^N$, where each pair consists of a visual stimulus represented by an RGB image $I^i \in \mathbb{R}^{H \times W \times 3}$ and a set of fixed length sequences $R_j^i \in \mathbb{R}^{L \times 2}$ of continuous eye movement trajectories collected from K^i different subjects for the i^{th} stimulus, our goal is to model the joint distribution of stimuli and trajectories. Each time step in R_j^i corresponds to a two-dimensional coordinate (x, y) , representing the spatial location of gaze at that timestep. For simplicity, we denote the set of stimuli as $\mathcal{S}$ and the set of eye movement trajectories as $\mathcal{R}$, and use R to refer to a single eye movement trajectory and I for a single visual stimulus. We model the distribution over $\mathcal{R}$ conditioned on $\mathcal{S}$ using a diffusion model. This framework enables learning the underlying generative process of eye movement behavior in response to complex visual stimuli, capturing both temporal dynamics and spatial variability. To achieve this, DiffEye adopts the denoising diffusion probabilistic (DDPM) model [56] to learn a distribution $p_\theta(\mathcal{R} \mid \mathcal{S})$ that approximates the true conditional distribution $q(\mathcal{R} \mid \mathcal{S})$ of eye-tracking data. DDPM consists of a forward process that progressively adds Gaussian noise to the trajectories over T_{diff} diffusion steps, and a reverse process that gradually denoises samples to recover realistic trajectories. The diffusion model is trained by minimizing the denoising objective $\min_\theta \mathbb{E}_{t_{\text{diff}}, R^{(0)}, \epsilon} [\|\epsilon - \epsilon_\theta(R^{(t_{\text{diff}})}, t_{\text{diff}}, I)\|^2]$, where $\epsilon \sim \mathcal{N}(0, \mathbf{I})$ is the ground-truth noise and $\epsilon_\theta(R^{(t_{\text{diff}})}, t_{\text{diff}}, I)$ is the model’s prediction of noise given the noised trajectory $R^{(t_{\text{diff}})}$, diffusion timestep t_{diff} , and conditioning image I .

3.2 Our Approach

Base Model We use a U-Net-based architecture [57] as the noise prediction model in our diffusion pipeline. The model consists of downsampling, mid, and upsampling blocks, where 1D convolution layers temporally downsample or upsample the trajectories while increasing or decreasing their number of features. Self-attention layers are applied after the convolutions to capture temporal dependencies in the eye-tracking data at multiple resolutions throughout the network. In addition to the original U-Net design, we incorporate the current diffusion timestep t_{diff} by passing it through a sinusoidal positional embedding module [58], followed by a fully connected layer, a SiLU activation [59], and another fully connected layer. The resulting timestep embedding is added to the output of the first convolution layer in each block, allowing the network to condition on t_{diff} and enabling effective diffusion training.

Stimuli Conditioning An essential component of our pipeline is conditioning the model on a given stimulus in the form of an image. To achieve this, first, the input trajectory is passed through a 1D convolution layer, projecting it from $R \in \mathbb{R}^{L \times 2}$ to $R_{\text{proj}} \in \mathbb{R}^{L \times D}$, where L is the number of trajectory timesteps and D is the embedding dimension.

Initially, we attempt to condition the model using the global feature vectors produced by the DI-NOv2 [60] Vision Transformer foundation model, chosen for its strong and well-established semantic representation capabilities. This global vector, originally of size $\mathbb{R}^{1 \times D_{\text{global}}}$, is projected via a fully connected layer to $\mathbb{R}^{1 \times D}$ and then concatenated with the projected trajectory tokens to form an input sequence $E_{\text{global}} \in \mathbb{R}^{(L+1) \times D}$. However, this approach alone leads to poor generation quality, as

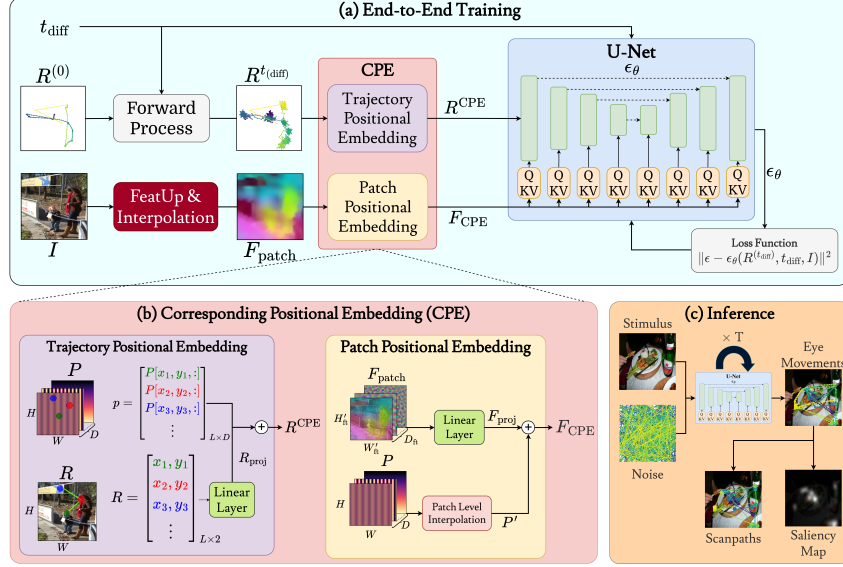


Figure 2: *An illustration of DiffEye. (a) End-to-end Training.* Given an initial trajectory $R^{(0)}$ and image I , noise is added to produce $R^{t(\text{diff})}$. FeatUp extracts patch features F_{patch} , and both inputs are passed to the **(b) CPE** module, which aligns trajectory and patch positions. The resulting representations R^{CPE} and F_{CPE} are processed by a U-Net with cross-attention at each block and optimized via diffusion loss. **(c) Inference.** Starting from noise, the model denoises for T steps to generate an eye movement trajectory, which can be used to produce scanpaths or saliency maps.

Table 1: *Training data comparison across methods.* DiffEye is trained solely on the MIT1003 dataset using full eye trajectory data, while other methods are typically trained on scanpaths and rely on larger or combined datasets.

Method	Datasets Used	# Eye Trajectories	# Scanpaths	# Images
IOR-ROI	OSIE [53]	-	8,400	560
DeepGazeIII	MIT1003 [18] and SALICON [43]	-	615,045	11,003
Chen et al.	AIR [62]	-	39,080	1,454
GazeFormer	COCO-Search18 [63]	-	62,020	6,202
HAT	COCO-Search18 [63], COCO-FreeView [64], MIT1003 [18], OSIE [53]	-	87,565	7,905
DiffEye	MIT1003 [18]	8,934	-	1,003

the global feature lacks localized spatial semantics necessary for effective conditioning. To better incorporate localized spatial information, we utilize patch-level features from DINOv2. Let the patch features be denoted as $F_{\text{patch}} \in \mathbb{R}^{H' \times W' \times D_{\text{feat}}}$, where (H', W') are the spatial dimensions of the patch grid. These features are projected via a linear transformation to $F_{\text{proj}} \in \mathbb{R}^{H' \times W' \times D}$. Rather than concatenating these directly with the trajectory, we adopt a cross-attention mechanism that enables interaction between the projected image patch tokens and the trajectory tokens $R_{\text{proj}} \in \mathbb{R}^{L \times D}$. Cross-attention is performed between the L trajectory tokens and the $N_p = H' \cdot W'$ image patch tokens (flattened from F_{proj}). Initial attempts with a single cross-attention layer before passing the trajectory to the U-Net proved insufficient for effective conditioning. To address this, we add cross-attention layers at the end of each U-Net block, which is empirically proven to enhance the performance. Additionally, to improve spatial precision, we replace DINOv2 patch embeddings with those from FeatUp, a model-agnostic framework that restores fine-grained spatial details in deep features [61]. FeatUp outputs a $224 \times 224 \times D_{\text{ft}}$ feature map, which we interpolate to $H'_{\text{ft}} \times W'_{\text{ft}} \times D_{\text{ft}}$, where $H'_{\text{ft}} = W'_{\text{ft}} = 32$ effectively doubling the spatial resolution relative to standard DINOv2 features.

Corresponding Positional Embedding (CPE) To further improve spatial alignment, we introduce a novel positional embedding method, *Corresponding Positional Embedding (CPE)*. Let the t_{diff} -step noisified input trajectory $R^{t(\text{diff})}$ be denoted as $R \in \mathbb{R}^{L \times 2}$, where each row represents a 2D coordinate (x_i, y_i) . We first project the trajectory into the hidden dimension, resulting in the embedded trajectory $R_{\text{proj}} \in \mathbb{R}^{L \times D}$. Next, we construct a 2D positional embedding grid $P \in \mathbb{R}^{H \times W \times D}$, based on the original image resolution (H, W) , using sinusoidal encodings [58]. For each timestep $i \in \{1, L\}$,

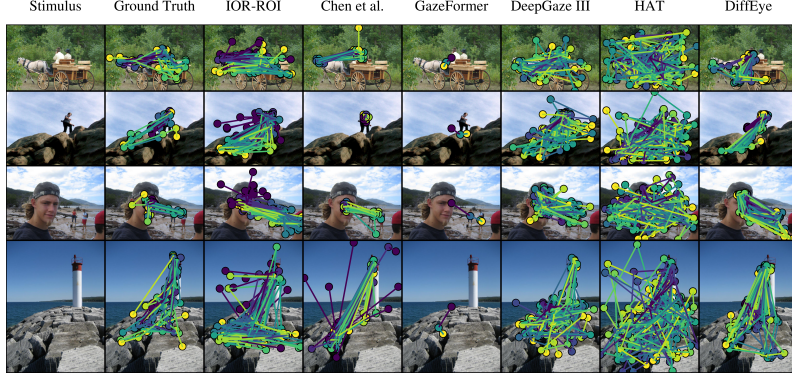


Figure 3: **Qualitative comparison of scanpath generation.** Scanpaths generated by DiffEye and baseline models are shown alongside ground truth annotations across four different scenes. Each row represents a unique stimulus, and each column shows the corresponding scanpaths generated from a specific method.

we retrieve the corresponding positional embedding vector from the grid as $p_i = P[y_i, x_i, :] \in \mathbb{R}^D$, where $R_i = (x_i, y_i)$ is the i -th coordinate in the original trajectory R . The final CPE-augmented trajectory token is then computed as:

$$R_i^{\text{CPE}} = R_{\text{proj}}[i] + p_i. \quad (1)$$

The same positional grid P is also applied to the image features. Let the projected image features be $F_{\text{proj}} \in \mathbb{R}^{H' \times W' \times D}$. We interpolate the positional grid P to the patch resolution, yielding $P' \in \mathbb{R}^{H' \times W' \times D}$. We then add this to the image features:

$$F_{\text{CPE}} = F_{\text{proj}} + P'. \quad (2)$$

By sharing and aligning positional information across both trajectories and image patches, CPE enables effective spatial correspondence and interaction between gaze behavior and visual content. Figure 2 summarizes our approach, including end-to-end training (a), the CPE module (b), and inference (c).

Data Preprocessing We use the MIT1003 dataset [18]² to train and evaluate our model. To the best of our knowledge, it is the only publicly available dataset that provides raw eye-tracking data for natural images obtained during a free-viewing task. The dataset contains recordings from 15 subjects who free-viewed 1,003 images for 3 seconds each, resulting in a total of 15,045 eye-tracking sequences. Data was collected using the ETL 400 ISCAN system at a sampling rate of 240 Hz. During preprocessing, we removed all blinks and NaN values, and retained only the sequences with at least 720 timesteps. After filtering, 8,934 trajectories remained. All remaining trajectories were truncated or downsampled to a uniform length of 720 samples, as our model is designed to generate fixed-length eye movement trajectories and the U-Net architecture requires several downsampling steps which is possible with a sequence length of 720. Additionally, all stimuli were resized to 224×224 pixels. The dataset is split into training (90%) and testing (10%) sets based on stimuli, ensuring that images used for evaluation were not seen during training.

4 Experimentation

Implementation Details We train our model using the Adam optimizer [65] for 3000 epochs with a fixed learning rate of 1×10^{-4} . During training, we utilize the DDPM scheduler with a linear noise schedule ranging from 1×10^{-4} to 2×10^{-2} . For sampling, we adopt Denoising Diffusion Implicit Models (DDIM) [66], which enable high-quality sample generation in significantly fewer steps without compromising performance. We set the number of diffusion steps to $T_{\text{diff}} = 1000$ during training and reduce it to 50 during sampling for improved efficiency. Throughout both training and inference, we apply classifier-free guidance (CFG) to effectively condition the model on the

²<https://people.csail.mit.edu/tjudd/WherePeopleLook/index.html>

Table 2: *Ablation study for scanpath and eye movement trajectory generation.* We evaluate the contribution of key components in DiffEye by removing modules such as FeatUp, CPE (Corresponding Positional Embedding), U-Net cross-attention, and patch-level features. Results are reported for both scanpath and continuous trajectory generation using four trajectory-based evaluation metrics.

Task	Configuration	Levenshtein Distance ($\times 10^2$) ↓		Discrete Fréchet Distance ($\times 10^2$) ↓		Dynamic Time Warping ($\times 10^4$) ↓		Time Delay Embedding ↓	
		Mean	Best	Mean	Best	Mean	Best	Mean	Best
Scanpath Generation	DiffEye	0.130	0.097	3.529	2.449	0.157	0.107	88.661	53.486
	w/o FeatUp	0.133	0.100	3.546	2.423	0.163	0.110	91.007	60.103
	w/o CPE	0.141	0.107	3.545	2.604	0.180	0.128	100.792	69.827
	w/o U-Net Cross-Attention	0.143	0.107	3.701	2.557	0.189	0.130	107.962	68.353
	w/o Patch Level Features	0.153	0.116	3.761	2.692	0.209	0.147	116.226	77.997
Eye Movement Trajectory Generation	DiffEye	10.083	8.289	3.601	2.460	11.834	8.212	35.228	20.968
	w/o FeatUp	10.265	8.736	3.844	2.623	12.513	8.645	41.224	26.453
	w/o CPE	10.773	9.200	3.621	2.599	13.430	10.068	44.403	28.904
	w/o U-Net Cross-Attention	10.971	9.394	3.828	2.587	14.716	10.992	56.739	38.264
	w/o Patch-Level Features	11.791	9.947	4.088	2.761	18.007	13.312	77.042	47.354

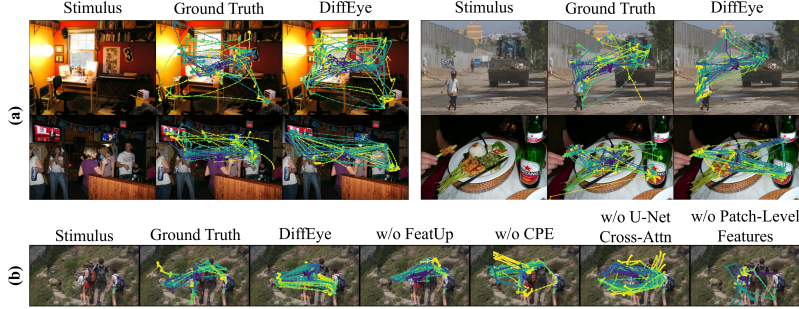


Figure 4: *Qualitative analysis and ablation study of continuous eye movement trajectory generation.* (a) Multiple eye movement trajectories generated by DiffEye alongside ground truth annotations across four different scenes. (b) Ablation study showing the impact of removing individual architectural components (FeatUp, CPE, cross-attention, and patch-level features) on continuous trajectory generation.

visual stimulus [67]. This encourages the model to learn both conditional and unconditional denoising behavior, supporting effective guidance at inference time. See Supplementary for the complete details. All experiments were conducted using an NVIDIA RTX A6000 GPU.

Baselines and Datasets We compare our method against several baseline models trained for the scanpath prediction/generations tasks. The baselines include the Human Attention Transformer (HAT) [20] and GazeFormer [22], both of which are transformer-based architectures, DeepGaze III [19], a convolutional network, Chen et al. [21], which is a reinforcement learning model that mainly focuses on visual question answering, and IOR-ROI [25]. Note that GazeFormer can not generate a distribution but can only predict scanpaths. DeepGaze III and IOR-ROI requires the number of fixations per scanpath as an input, which we set as 10 for all experiments. Each baseline is used to generate scanpaths for all test stimuli. Scanpath baselines were trained on scanpaths coming from various combinations of datasets as shown in Table 1. In contrast, our method is trained solely on a subset of the raw eye movement trajectories from the MIT1003 dataset which is relatively very small dataset with fewer pairs of stimuli and eye movement trajectories, yet achieves competitive performance, demonstrating robustness under limited data. We evaluate our results on the test sets of the MIT1003 and OSIE [53]³ datasets. It is important to note that the MIT1003 dataset does not provide official train-test splits; therefore, we created a random split of 100 images with eye-tracking data from 15 subjects each as described in Section 3.2. As a result, methods such as HAT and DeepGazeIII, whose training data included MIT1003, have already been exposed to the evaluation data. The OSIE test set contains 70 images with eye-tracking data from 15 subjects.

Evaluation Metrics For scanpath generation, we follow the evaluation protocol proposed in [33, 23]. For each image in the test set, we generate 15 eye-tracking trajectories per model, extract fixation points using the conversion method described in [18], and construct the corresponding scanpaths. For models supporting multiple visual tasks, we ensured that scanpaths were generated specifically for the Free-Viewing task. We evaluate the results using four standard metrics commonly used in

³<https://github.com/chenxy99/Scanpaths/tree/main/OSIE>

Table 3: *Quantitative comparison of scanpath generation on the MIT1003 and OSIE datasets.* We report the performance of each method using four commonly used trajectory-based metrics: Levenshtein Distance, Discrete Fréchet Distance, Dynamic Time Warping (DTW), and Time Delay Embedding (TDE). The label (seen) indicates that the model’s training set includes the test set images. **Bold** values indicate the best scores, while underlined values denote the second-best scores.

Test Dataset	Method	Levenshtein Distance ↓		Discrete Fréchet Distance ↓ ($\times 10^2$)		Dynamic Time Warping ↓ ($\times 10^3$)		Time Delay Embedding ↓	
		Mean	Best	Mean	Best	Mean	Best	Mean	Best
MIT1003	IOR-ROI	<u>13.574</u>	<u>11.092</u>	3.777	2.460	1.834	1.317	108.284	80.944
	DeepGaze III (seen)	14.415	11.856	3.553	2.160	<u>1.757</u>	<u>1.141</u>	96.456	<u>65.408</u>
	Chen et al.	14.874	12.943	<u>3.704</u>	2.602	1.851	1.409	<u>92.100</u>	74.212
	GazeFormer	-	12.614	-	3.553	-	1.545	-	93.751
	HAT (seen)	18.440	14.645	4.293	2.940	2.680	1.862	131.516	97.232
	DiffEye	13.009	9.709	3.529	<u>2.449</u>	1.573	1.067	88.661	53.486
OSIE	IOR-ROI	<u>14.836</u>	<u>12.152</u>	3.357	<u>2.228</u>	<u>1.699</u>	1.167	92.960	70.624
	DeepGaze III	15.507	12.532	<u>3.206</u>	2.077	1.765	<u>1.166</u>	84.337	<u>57.786</u>
	Chen et al.	17.024	14.910	3.275	2.290	1.772	1.276	78.286	61.509
	GazeFormer	-	15.320	-	3.257	-	1.687	-	81.878
	HAT	19.419	15.607	3.712	2.598	2.501	1.757	111.413	83.140
	DiffEye	14.771	12.077	3.068	2.238	1.552	1.089	<u>81.925</u>	54.347

the eye-tracking literature: *Levenshtein Distance*, which measures the sequence similarity based on insertion, deletion, and substitution operations; *Discrete Fréchet Distance (DFD)*, which captures the similarity between curves while considering their sequential nature; *Dynamic Time Warping (DTW)*, which aligns sequences that may vary in speed or length; and *Time Delay Embedding (TDE)*, which assesses the temporal structure of trajectories [68]. Each test image is associated with 15 ground truth scanpaths. We generate 15 scanpaths per model; for deterministic models such as Gazeformer, only a single prediction is evaluated. For each metric, we report two scores: *best* and *mean*. The *best* score is obtained by computing the evaluation metric between each ground truth scanpath and all generated scanpaths, selecting the most similar (or least distant) one, and then averaging over all scanpaths and images. The *mean* score is obtained by averaging the metric across all pairwise comparisons between ground truth and generated scanpaths for each image, followed by averaging across all test images (see Algorithm 1 in Supplementary.).

Scanpath Generation Results As shown in Table 3, the *mean* scores highlight our model’s superior ability to generate trajectories that align closely with the overall distribution of human fixations. This is demonstrated by the lowest Levenshtein and TDE values, which capture spatial and temporal deviations across full scanpaths. Furthermore, the DTW and DFD metric, sensitive to the shape and curvature of the trajectories, indicates that our model better preserves the sequential flow of eye movements. Figure 3 visually reinforces these findings. In the second row (the image of the climbing human), the model by Chen et al. overly concentrates predictions around the low-resolution human head, while DiffEye distributes gaze more naturally across salient regions, including the background, showing stronger alignment with the ground truth. In contrast, DeepGaze III and HAT produce diffuse trajectories, suggesting a failure to learn coherent attention patterns. IOR-ROI performs slightly better, but still deviates from the true distribution. Gazeformer, meanwhile, does not generate a distribution, but performs prediction, resulting in higher Levenshtein and TDE mean scores, indicating weaker alignment with human gaze distributions. Note that our model is trained on the MIT1003 dataset and evaluated on both the MIT1003 test set and the entirely unseen OSIE dataset, demonstrating strong generalization. Additional qualitative results are included in Supplementary.

Ablation Study We conduct an ablation study to assess the importance of key architectural components in DiffEye. Specifically, we compare the full model, comprising FeatUp high-resolution patch features, the proposed CPE, and cross-attention at all U-Net layers, against variants where one of these components was removed. As shown in Table 2, ablating any of these components degrades performance across both scanpath and continuous eye movement trajectory tasks. Removing FeatUp and replacing it with DINOv2 features leads to a consistent drop in performance. This confirms the benefit of FeatUp, which provides higher resolution and spatially structured visual features essential for precise modeling. The CPE module, a novel contribution of our work, also significantly boosts performance. By aligning eye trajectories with the spatial layout of the visual stimulus, CPE enhances the model’s ability to localize attention signals effectively, especially evident in the Levenshtein and Fréchet distances, which are sensitive to spatial alignment. We further evaluate the effect of distributing cross-attention across all U-Net blocks versus using a single cross-attention layer at the beginning. Disabling full cross-attention notably worsens performance, suggesting that deep conditioning through the network is crucial to preserve and propagate stimulus-related signals.

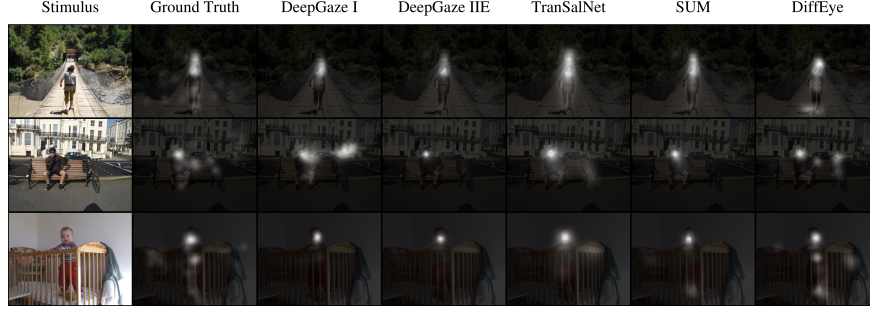


Figure 5: *Qualitative comparison of saliency map predictions.* Saliency maps generated by DiffEye and baseline models are shown alongside ground truth maps for four different scenes. Each row corresponds to a different stimulus, with columns displaying the stimulus, ground truth, and predictions.

Finally, we test using only a global token instead of patch-level features which is concatenated as an additional token to the trajectory tokens before passing it through the U-Net. This setup parallels DiffGaze [23], a model designed for 360° images, which also uses a conditioning mechanism with global token. Our results show that such global conditioning is insufficient for natural images, where fine-grained patch-level information is needed to localize saliency and highlight the value of our patch-based strategy. Please refer to Figure 4 for qualitative comparisons that further illustrate the contributions of each component. See Supplementary for additional qualitative results for the eye movement trajectory generation.

Saliency Prediction We compare our method against the SUM model [50], TranSalNet [49], DeepGaze I [47], and DeepGaze IIE [48] which are all trained specifically for the saliency prediction task. Each baseline model is used to generate saliency maps for the MIT1003 test split used to evaluate scanpaths. We convert the continuous eye movement trajectories generated from DiffEye to saliency maps as described in the Supplementary materials. Figure 5 presents qualitative results for the saliency prediction task comparing DiffEye with baseline models. Despite not being explicitly trained for saliency map generation, our model produces visually competitive outputs. For instance, in the second row, while DeepGaze IIE and SUM primarily focus on the subject’s face, DiffEye captures a broader set of salient regions, including the face, the bicycle, and the left hand, closely resembling the ground truth distribution. These results demonstrate our model’s capacity to generalize beyond its original training objective. Additional evaluations and qualitative results are provided in Supplementary.

5 Discussion & Limitations

We present DiffEye, a novel diffusion-based model for generating eye movement trajectories on natural images. Unlike prior methods that rely on deterministic architectures, autoregressive sampling, and training with compressed representations like saliency maps or discrete scanpaths, DiffEye captures the variability of human visual behavior using state-of-the-art generative modeling and trains directly on raw eye-tracking data. To support this, we introduce CPE, a mechanism that aligns spatial gaze patterns with semantic content via cross-attention between image patches and trajectory timesteps. By modeling scanpaths as a generative process, DiffEye produces realistic and diverse gaze behaviors and achieves state-of-the-art results. Beyond technical performance, it has potential applications in developmental science by simulating population-specific gaze patterns. This may support the diagnosis of conditions such as autism by generating stimuli that maximize group-level distinctions in eye-tracking studies. Although DiffEye achieves strong results using only the MIT1003 dataset—the only known dataset offering raw eye trajectories alongside scanpaths for natural images—the dataset size remains a key limitation and scaling its performance will require access to additional data. Future work will explore transfer learning from datasets containing saliency maps and scanpaths, and aim to include data from both neurotypical and developmentally diverse populations. We also advocate for the release of raw eye-tracking data. At present, DiffEye generates only fixed-length outputs at 240 Hz; broader data availability would enable support for variable-length sequences and diverse sampling rates. Ultimately, DiffEye could also be used to synthesize training data, helping to address data scarcity in this field.

Acknowledgments and Disclosure of Funding

Portions of this research were supported in part by the Health Care Engineering Systems Center in the Grainger College of Engineering at UIUC, and the National Institutes of Health (NIH) under award P41EB028242.

References

- [1] Alfred L Yarbus. *Eye movements and vision*. Springer, 2013.
- [2] Constantin A. Rothkopf, Dana H. Ballard, and Mary M. Hayhoe. Task and context determine where you look. *Journal of Vision*, 7(14):16, July 2016.
- [3] Viviane Clay, Peter König, and Sabine U. König. Eye tracking in virtual reality. *Journal of Eye Movement Research*, 12(1), April 2019.
- [4] Rehman Razzak, Yi (Joy) Li, Jing (Selena) He, Sungchul Jung, Chao Mei, and Yan Huang. Using virtual reality to enhance attention for autistic spectrum disorder with eye tracking. *High-Confidence Computing*, 5(1):100234, March 2025.
- [5] Xiaojun Ren, Jiluan Fan, Ning Xu, Shaowei Wang, Changyu Dong, and Zikai Wen. Dpgazesynth: Enhancing eye-tracking virtual reality privacy with differentially private data synthesis. *Information Sciences*, 675:120720, July 2024.
- [6] Jihwan Kim, Jejoong Kim, Myeongul Jung, Taesoo Kwon, and Kwanguk Kenny Kim. Individualized foveated rendering with eye-tracking head-mounted display. *Virtual Reality*, 28(1), January 2024.
- [7] Wenxuan Liu, Budmonde Duinkharjav, Qi Sun, and Sai Qian Zhang. Fovealnet: Advancing ai-driven gaze tracking solutions for efficient foveated rendering in virtual reality. *IEEE Transactions on Visualization and Computer Graphics*, page 1–11, 2025.
- [8] Jiaying Liu, Joe Phua, Dean Krugman, Linjia Xu, Glen Nowak, and Lucy Popova. Do young adults attend to health warnings in the first iqos advertisement in the u.s.? an eye-tracking approach. *Nicotine & Tobacco Research*, 23(5):815–822, November 2020.
- [9] Xuebai Zhang and Shyan-Ming Yuan. An eye tracking analysis for video advertising: Relationship between advertisement elements and effectiveness. *IEEE Access*, 6:10699–10707, 2018.
- [10] Kenneth Holmqvist, Marcus Nyström, Richard Andersson, Richard Dewhurst, Halszka Jarodzka, and Joost Van de Weijer. *Eye tracking: A comprehensive guide to methods and measures*. oup Oxford, 2011.
- [11] Kritika Nayar, Frederick Shic, Molly Winston, and Molly Losh. A constellation of eye-tracking measures reveals social attention differences in asd and the broad autism phenotype. *Molecular autism*, 13(1):18, 2022.
- [12] Ryan Anthony de Belen, Hannah Pincham, Antoinette Hodge, Natalie Silove, Arcot Sowmya, Tomasz Bednarz, and Valsamma Eapen. Eye-tracking correlates of response to joint attention in preschool children with autism spectrum disorder. *BMC psychiatry*, 23(1):211, 2023.
- [13] James C. McPartland, Sara Jane Webb, Brandon Keehn, and Geraldine Dawson. Patterns of visual attention to faces and objects in autism spectrum disorder. *Journal of Autism and Developmental Disorders*, 41(2):148–157, May 2010.
- [14] Chawarska Katarzyna, Volkmar Fred, and Klin Ami. Limited attentional bias for faces in toddlers with autism spectrum disorders. *Archives of general psychiatry*, 67(2):178–185, 2010.
- [15] Zoya Bylinskii, Adrià Recasens, Ali Borji, Aude Oliva, Antonio Torralba, and Frédo Durand. Where should saliency models look next? In *Computer Vision—ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part V 14*, pages 809–824. Springer, 2016.
- [16] Ali Borji. Saliency prediction in the deep learning era: Successes and limitations. *IEEE transactions on pattern analysis and machine intelligence*, 43(2):679–700, 2019.
- [17] Jason W. Griffin, Adam Naples, Raphael Bernier, Katarzyna Chawarska, Geraldine Dawson, James Dziura, Susan Faja, Shafali Jeste, Natalia Kleinhans, Catherine Sugar, Sara Jane Webb, Frederick Shic, and James C. McPartland. Spatiotemporal eye movement dynamics reveal altered face prioritization in early visual processing among autistic children. *Biological Psychiatry: Cognitive Neuroscience and Neuroimaging*, 10(1):45–57, January 2025.

- [18] Tilke Judd, Krista Ehinger, Frédo Durand, and Antonio Torralba. Learning to predict where humans look. In *IEEE International Conference on Computer Vision (ICCV)*, 2009.
- [19] Matthias Kümmeler, Matthias Bethge, and Thomas S. A. Wallis. Deepgaze iii: Modeling free-viewing human scanpaths with deep learning. *Journal of Vision*, 22(5):7, April 2022.
- [20] Zhibo Yang, Sounak Mondal, Seoyoung Ahn, Ruoyu Xue, Gregory Zelinsky, Minh Hoai, and Dimitris Samaras. Unifying top-down and bottom-up scanpath prediction using transformers. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, page 1683–1693. IEEE, June 2024.
- [21] Xianyu Chen, Ming Jiang, and Qi Zhao. Predicting human scanpaths in visual question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10876–10885, 2021.
- [22] Sounak Mondal, Zhibo Yang, Seoyoung Ahn, Dimitris Samaras, Gregory Zelinsky, and Minh Hoai. Gazeformer: Scalable, effective and fast prediction of goal-directed human attention. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, page 1441–1450. IEEE, June 2023.
- [23] Chuhan Jiao, Yao Wang, Guanhua Zhang, Mihai Băce, Zhiming Hu, and Andreas Bulling. Diffgaze: A diffusion model for continuous gaze sequence generation on 360° images, 2024.
- [24] Matthias Kümmeler and Matthias Bethge. State-of-the-art in human scanpath prediction. *arXiv preprint arXiv:2102.12239*, 2021.
- [25] Wanjie Sun, Zhenzhong Chen, and Feng Wu. Visual scanpath prediction using ior-roi recurrent mixture density network. *IEEE transactions on pattern analysis and machine intelligence*, 43(6):2101–2118, 2019.
- [26] Marc Assens, Xavier Giro-i Nieto, Kevin McGuinness, and Noel E O’Connor. Pathgan: Visual scanpath prediction with generative adversarial networks. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, pages 0–0, 2018.
- [27] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.
- [28] Shima Rashidi, Krista Ehinger, Andrew Turpin, and Lars Kulik. Optimal visual search based on a model of target detectability in natural images. *Advances in Neural Information Processing Systems*, 33:9288–9299, 2020.
- [29] Zhibo Yang, Sounak Mondal, Seoyoung Ahn, Gregory Zelinsky, Minh Hoai, and Dimitris Samaras. Target-absent human attention. In *European Conference on Computer Vision*, pages 52–68. Springer, 2022.
- [30] Marc Assens Reina, Xavier Giro-i Nieto, Kevin McGuinness, and Noel E O’Connor. Saltinet: Scan-path prediction on 360 degree images using saliency volumes. In *Proceedings of the IEEE international conference on computer vision workshops*, pages 2331–2338, 2017.
- [31] Rong Quan, Yantao Lai, Mengyu Qiu, and Dong Liang. Pathformer3d: A 3d scanpath transformer for 360 images. In *European Conference on Computer Vision*, pages 73–90. Springer, 2024.
- [32] Daniel Martin, Ana Serrano, Alexander W Bergman, Gordon Wetzstein, and Belen Masia. Scangan360: A generative model of realistic scanpaths for 360 images. *IEEE Transactions on Visualization and Computer Graphics*, 28(5):2003–2013, 2022.
- [33] Yujia Wang, Fang-Lue Zhang, and Neil A Dodgson. Scantd: 360° scanpath prediction based on time-series diffusion. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 7764–7773, 2024.
- [34] Xianyu Chen, Ming Jiang, and Qi Zhao. Beyond average: Individualized visual scanpath prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 25420–25431, 2024.
- [35] Anne M. Treisman and Garry Gelade. A feature-integration theory of attention. *Cognitive Psychology*, 12(1):97–136, January 1980.
- [36] Christof Koch and Shimon Ullman. *Shifts in Selective Visual Attention: Towards the Underlying Neural Circuitry*, page 115–141. Springer Netherlands, 1987.
- [37] John M Findlay and Iain D Gilchrist. Visual attention: The active vision perspective. In *Vision and attention*, pages 83–103. Springer, 2001.

- [38] Gregory J Zelinsky. A theory of eye movements during target acquisition. *Psychological review*, 115(4):787, 2008.
- [39] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(11):1254–1259, 1998.
- [40] David J Berg, Susan E Boehnke, Robert A Marino, Douglas P Munoz, and Laurent Itti. Free viewing of dynamic stimuli by humans and monkeys. *Journal of vision*, 9(5):19–19, 2009.
- [41] Ali Borji and Laurent Itti. State-of-the-art in visual attention modeling. *IEEE transactions on pattern analysis and machine intelligence*, 35(1):185–207, 2012.
- [42] Yin Li, Xiaodi Hou, Christof Koch, James M. Rehg, and Alan L. Yuille. The secrets of salient object segmentation. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 280–287, 2014.
- [43] Ming Jiang, Shengsheng Huang, Juanyong Duan, and Qi Zhao. Salicon: Saliency in context. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1072–1080, 2015.
- [44] Saumya Jetley, Naila Murray, and Eleonora Vig. End-to-end saliency mapping via probability distribution prediction. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5753–5761, 2016.
- [45] Srinivas SS Kruthiventi, Kumar Ayush, and R Venkatesh Babu. Deepfix: A fully convolutional neural network for predicting human eye fixations. *IEEE Transactions on Image Processing*, 26(9):4446–4456, 2017.
- [46] Marcella Cornia, Lorenzo Baraldi, Giuseppe Serra, and Rita Cucchiara. Predicting human eye fixations via an lstm-based saliency attentive model. *IEEE Transactions on Image Processing*, 27(10):5142–5154, 2018.
- [47] Matthias Kümmerer, Lucas Theis, and Matthias Bethge. Deep gaze i: Boosting saliency prediction with feature maps trained on imagenet. In *ICLR (Workshop)*, 2015.
- [48] Akis Linardos, Matthias Kümmerer, Ori Press, and Matthias Bethge. Deepgaze iie: Calibrated prediction in and out-of-domain for state-of-the-art saliency modeling. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 12899–12908, 2021.
- [49] Jianxun Lou, Hanhe Lin, David Marshall, Dietmar Saupe, and Hantao Liu. Transalnet: Towards perceptually relevant visual saliency prediction. *Neurocomputing*, 494:455–467, July 2022.
- [50] Alireza Hosseini, Amirhossein Kazerooni, Saeed Akhavan, Michael Brudno, and Babak Taati. Sum: Saliency unification through mamba for visual attention modeling. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, page 1597–1607. IEEE, February 2025.
- [51] Dario D Salvucci and Joseph H Goldberg. Identifying fixations and saccades in eye-tracking protocols. In *Proceedings of the 2000 symposium on Eye tracking research & applications*, pages 71–78, 2000.
- [52] Ali Borji and Laurent Itti. Cat2000: A large scale fixation dataset for boosting saliency research. *arXiv preprint arXiv:1505.03581*, 2015.
- [53] Juan Xu, Ming Jiang, Shuo Wang, Mohan S Kankanhalli, and Qi Zhao. Predicting human gaze beyond pixels. *Journal of vision*, 14(1):28–28, 2014.
- [54] Lai Jiang, Yifei Li, Shengxi Li, Mai Xu, Se Lei, Yichen Guo, and Bo Huang. Does text attract attention on e-commerce images: A novel saliency prediction dataset and method. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2088–2097, 2022.
- [55] Yue Jiang, Luis A Leiva, Hamed Rezazadegan Tavakoli, Paul RB Houssel, Julia Kylmälä, and Antti Oulasvirta. Ueyes: understanding visual saliency across user interface types. In *Proceedings of the 2023 CHI conference on human factors in computing systems*, pages 1–21, 2023.
- [56] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 6840–6851. Curran Associates, Inc., 2020.
- [57] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. *U-Net: Convolutional Networks for Biomedical Image Segmentation*, page 234–241. Springer International Publishing, 2015.

- [58] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.
- [59] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.
- [60] 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. Featured Certification.
- [61] Stephanie Fu, Mark Hamilton, Laura E. Brandt, Axel Feldmann, Zhoutong Zhang, and William T. Freeman. Featup: A model-agnostic framework for features at any resolution. In *The Twelfth International Conference on Learning Representations*, 2024.
- [62] Shi Chen, Ming Jiang, Jinhui Yang, and Qi Zhao. Air: Attention with reasoning capability. In *ECCV*, 2020.
- [63] Yupei Chen, Zhibo Yang, Seoyoung Ahn, Dimitris Samaras, Minh Hoai, and Gregory Zelinsky. Cocossearch18 fixation dataset for predicting goal-directed attention control. *Scientific reports*, 11(1):8776, 2021.
- [64] Yupei Chen, Zhibo Yang, Souradeep Chakraborty, Sounak Mondal, Seoyoung Ahn, Dimitris Samaras, Minh Hoai, and Gregory Zelinsky. Characterizing target-absent human attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5031–5040, 2022.
- [65] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- [66] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021.
- [67] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021.
- [68] Ramin Fahimi and Neil D. B. Bruce. On metrics for measuring scanpath similarity. *Behavior Research Methods*, 53(2):609–628, August 2020.
- [69] Zoya Bylinskii, Tilke Judd, Aude Oliva, Antonio Torralba, and Fredo Durand. What do different evaluation metrics tell us about saliency models? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(3):740–757, March 2019.

NeurIPS Paper Checklist

1. Claims

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

Answer: [\[Yes\]](#)

Justification: We have explicitly explained our paper's contribution and scope in the Abstract and Section 1.

Guidelines:

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

2. Limitations

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

Answer: [\[Yes\]](#)

Justification: Our limitations are discussed in the paper in Section 5.

Guidelines:

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

3. Theory Assumptions and Proofs

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

Answer: [\[NA\]](#)

Justification: Our paper does not include theoretical results.

Guidelines:

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

4. Experimental Result Reproducibility

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

Answer: [\[Yes\]](#)

Justification: We have describe all information needed in to reproduce the experimental results in Sections [3.2](#), [4](#).

Guidelines:

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

5. Open access to data and code

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

Answer: [Yes]

Justification: We are going to release the code upon acceptance of the paper. The datasets we used are publicly available and we describe how we used them in Section 3.2.

Guidelines:

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

6. Experimental Setting/Details

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

Answer: [Yes]

Justification: Implementation and dataset details are described in Sections 4.

Guidelines:

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

7. Experiment Statistical Significance

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

Answer: [No]

Justification: To compute error bars, we would need to perform computationally intensive simulations.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. Experiments Compute Resources

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

Answer: [Yes]

Justification: We discuss the compute resources used in Section 4.

Guidelines:

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

9. Code Of Ethics

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

Answer: [Yes]

Justification: It conforms the Code of Ethics.

Guidelines:

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

10. Broader Impacts

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

Answer: [Yes]

Justification: The main motivation of this work is for the use of this technology to better diagnose and understand developmental conditions such autism and this is discussed in Sections 1, 5.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. Safeguards

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

Answer: [NA]

Justification: This work does not pose such risks.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. Licenses for existing assets

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

Answer: [Yes]

Justification: We properly credited the owners of the assets.

Guidelines:

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

- If this information is not available online, the authors are encouraged to reach out to the asset’s creators.

13. **New Assets**

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

Answer: [NA]

Justification: We do not release new assets.

Guidelines:

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

14. **Crowdsourcing and Research with Human Subjects**

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

Answer: [NA]

Justification: This work does not involve crowdsourcing nor research with human subjects.

Guidelines:

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

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

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

Answer: [NA]

Justification: This work does not involve crowdsourcing nor research with human subjects.

Guidelines:

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

16. **Declaration of LLM usage**

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

Answer: [NA]

Justification: The core method does not involve any LLMs.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (<https://neurips.cc/Conferences/2025/LLM>) for what should or should not be described.

A Appendix / supplemental material

A.1 Classifier Free Guidance

Given a noised eye movement trajectory $R^{(t_{\text{diff}})}$ at diffusion timestep t_{diff} , we perform two forward passes through the noise prediction network ϵ_θ : one conditioned on the visual stimulus I , yielding $\epsilon_\theta(R^{(t_{\text{diff}})}, t_{\text{diff}}, I)$, and another using an unconditional input I_{uc} , defined as a zero matrix, yielding $\epsilon_\theta(R^{(t_{\text{diff}})}, t_{\text{diff}}, I_{\text{uc}})$. The final guided noise prediction $\hat{\epsilon}_\theta$ is computed as:

$$\hat{\epsilon}_\theta = (1 - c) \cdot \epsilon_\theta(R^{(t_{\text{diff}})}, t_{\text{diff}}, I_{\text{uc}}) + c \cdot \epsilon_\theta(R^{(t_{\text{diff}})}, t_{\text{diff}}, I), \quad (3)$$

where c is the classifier-free guidance scale, which we set to 4 during inference. To enable this, we simulate the unconditional setting during training by randomly replacing the conditioning input I with a zero matrix in 10% of the training samples. This encourages the model to learn both conditional and unconditional denoising behavior, supporting effective guidance at inference time.

A.2 Evaluation Algorithm

Below is the algorithm we used to compute the *mean* and *best* scores for each of the scanpath and continuous trajectory metrics.

Algorithm 1 Evaluation of Scanpath and Continuous Trajectory Generation Metrics

Require: Test set of images $\mathcal{I}$; ground truth scanpaths $\mathcal{G}_i = \{g_1, \dots, g_N\}$; generated scanpaths $\mathcal{S}_i = \{s_1, \dots, s_M\}$; evaluation metric $d(\cdot, \cdot)$
Ensure: Overall *best* and *mean* scores across test images
1: Initialize `best_scores` $\leftarrow []$ and `mean_scores` $\leftarrow []$
2: **for** each image $i \in \mathcal{I}$ **do**
3: Initialize `image_best` $\leftarrow []$ and `image_mean` $\leftarrow []$
4: **for** each $g \in \mathcal{G}_i$ **do**
5: Compute distances $\{d(g, s) \mid s \in \mathcal{S}_i\}$
6: `best_g` $\leftarrow \min_{s \in \mathcal{S}_i} d(g, s)$
7: `mean_g` $\leftarrow \frac{1}{M} \sum_{s \in \mathcal{S}_i} d(g, s)$
8: Append `best_g` to `image_best`
9: Append `mean_g` to `image_mean`
10: **end for**
11: Append $\frac{1}{N} \sum \text{image_best}$ to `best_scores`
12: Append $\frac{1}{N} \sum \text{image_mean}$ to `mean_scores`
13: **end for**
14: **return** Overall Best = $\frac{1}{|\mathcal{I}|} \sum \text{best_scores}$, Overall Mean = $\frac{1}{|\mathcal{I}|} \sum \text{mean_scores}$

A.3 Additional Scanpath Distributions and Evaluations

Please see additional examples of scanpaths generated by DiffEye for the MIT1003 dataset in Fig. 6.

In addition to the trajectory-based metrics, we evaluate our model using Sequence Score (SS) and Semantic Sequence Score (Sem SS). The results, presented in Table 4, show that our model, DiffEye, achieves state-of-the-art performance on the MIT1003 dataset, outperforming all baselines in both SS and Sem SS. Furthermore, our model generalizes effectively by remaining highly competitive on the unseen OSIE dataset. This robust performance is particularly notable given that DiffEye was trained on significantly less data than competing models. For instance, while DeepGazeIII was trained on approximately 600,000 scanpaths, our method achieved these results with only 8,900 trajectories, demonstrating our approach’s ability to produce high-quality scanpaths by leveraging rich, raw trajectory information.

A.4 Additional Continuous Eye Movement Trajectory Distributions

Please see additional examples of continuous eye movement trajectories generated by DiffEye for the MIT1003 dataset in Fig. 7.

A.5 Analysis of Saliency Prediction

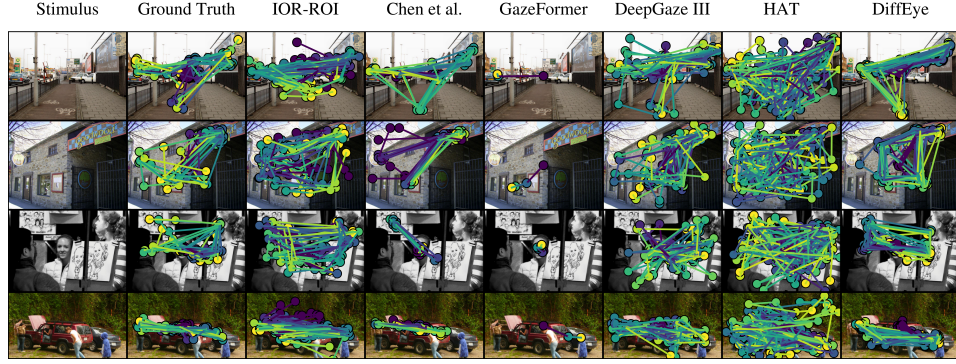


Figure 6: **Additional scanpath generation results.** Scanpaths generated by DiffEye and baseline models are shown alongside ground truth annotations across four different scenes. Each row represents a unique stimulus, and each column shows the generated scanpaths for each method.

Table 4: **Quantitative comparison using Sequence Score (SS) and Semantic Sequence Score (Sem SS) on the MIT1003 and OSIE datasets.** We report the mean performance for each method. **Bold** values indicate the best scores, while underlined values denote the second-best scores.

Model	MIT1003		OSIE	
	SS ($\uparrow$)	Sem SS ($\uparrow$)	SS ($\uparrow$)	Sem SS ($\uparrow$)
DiffEye (Ours)	0.4782	0.6611	0.4371	0.5837
HAT	0.4079	0.5794	0.4002	0.5791
Gazeformer	0.3531	0.4522	0.2713	0.3602
DeepGazeIII	0.4440	<u>0.6604</u>	0.4623	0.6459
ROI	<u>0.4506</u>	0.6603	<u>0.4404</u>	<u>0.6110</u>
Chen et al.	<u>0.4237</u>	0.6397	0.4333	0.5711

Saliency Prediction To evaluate the spatial realism of our generated eye-tracking sequences, we compared our method against several models trained specifically for the saliency prediction task. The baselines include: the SUM model [50], which integrates the Mamba architecture with a U-Net to output saliency maps across diverse image types; TranSalNet [49], which leverages transformers for saliency prediction; and DeepGaze I [47] and DeepGaze IIE [48], which are based on pretrained convolutional neural networks.

Table 5: Saliency prediction comparison. Bold is best and underline is second best.

Method	AUC-Judd $\uparrow$	AUC-Borji $\uparrow$	NSS $\uparrow$	SIM $\uparrow$	CC $\uparrow$	KL $\downarrow$
DeepGaze I	0.883	0.766	2.306	0.484	0.580	<u>0.980</u>
DeepGaze IIE	<u>0.923</u>	0.830	<u>3.321</u>	0.618	0.794	0.552
TranSalNet	0.896	0.874	2.443	0.508	0.658	6.369
SUM	0.931	<u>0.8458</u>	3.611	0.727	0.878	1.438
DiffEye	0.832	0.737	1.991	0.447	0.527	1.991

For each image in the test set, we generated 15 eye-tracking trajectories using our method. From each trajectory, we extracted fixation points using a script provided by [18], and used these to create individual fixation maps. These maps were aggregated and convolved with a Gaussian kernel to produce a single saliency map per image. For the baseline methods, which directly output saliency maps, we passed the same test images through each model. We then compared all predicted saliency maps, including ours, to the ground truth saliency maps provided in the dataset using six standard metrics: AUC-Judd, AUC-Borji, Normalized Scanpath Saliency (NSS), Similarity (SIM), Pearson’s Correlation Coefficient (CC), and Kullback–Leibler Divergence (KL). Please refer to [69] for a comprehensive detailing of the saliency metrics used. Fig. 8 shows additional examples of saliency maps generated by DiffEye and the baselines for the MIT1003 dataset and Table 5 reports the quantitative results.

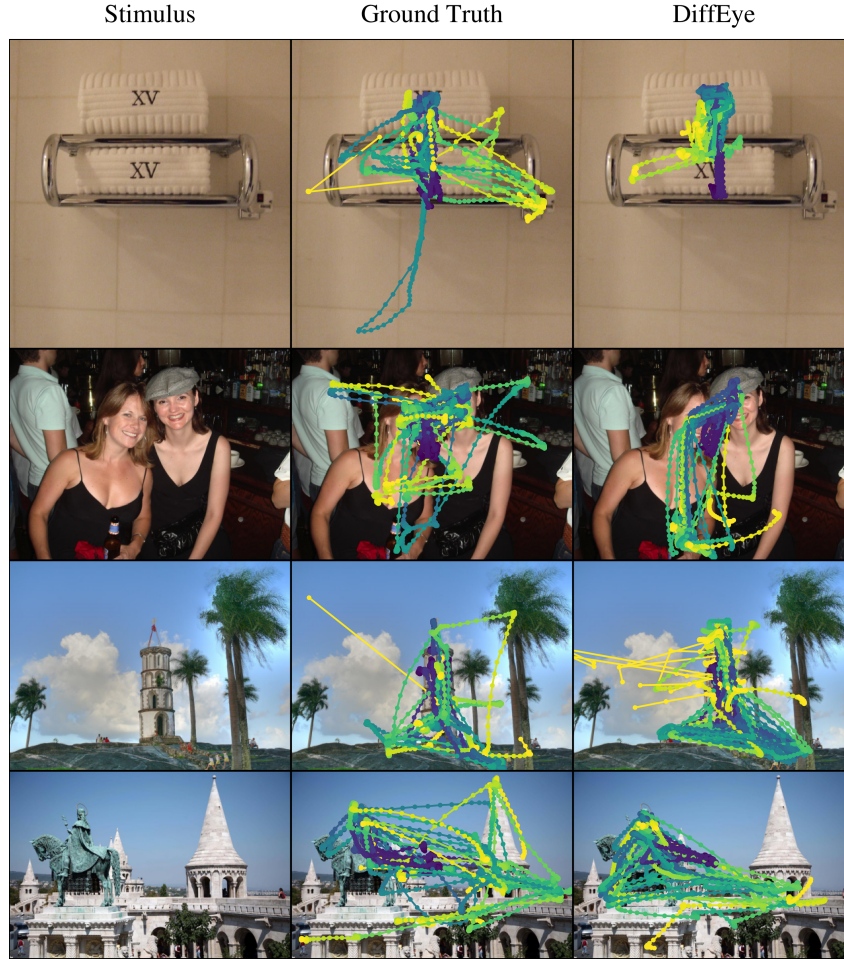


Figure 7: **Additional qualitative results of continuous eye movement trajectory generation.** Additional eye movement trajectories generated by DiffEye alongside ground truth annotations across four different scenes.



Figure 8: **Additional qualitative results for saliency prediction.** Saliency maps generated by DiffEye and baseline models are shown alongside ground truth maps for four different scenes. Each row corresponds to a different stimulus image, with columns displaying the stimulus, ground truth saliency map, and predictions.

A.6 Analysis of Statistical Properties of Scanpath Generation

To evaluate the plausibility of the generated scanpaths, we compare their statistical properties against those of ground truth human eye movements. Figure 9 shows the distributions of three key metrics: saccade amplitude, saccade direction, and inter-saccade angle for both the MIT1003 and OSIE datasets.

Our model’s performance (blue line) demonstrates a strong alignment with the ground truth distributions (black dashed line) across all three metrics. In the saccade amplitude distribution, our model successfully captures the peak at lower pixel values. For saccade direction, our model accurately reflects the horizontal bias present in human vision (peaks at 0 and ± 180 degrees). Finally, in the inter-saccade angle distribution, our model correctly shows a strong tendency for forward movements (peak near 0 degrees) and return saccades (smaller peak near ± 180 degrees). These results are consistent across both datasets, confirming that our approach generates statistically more realistic scanpaths than the compared methods.

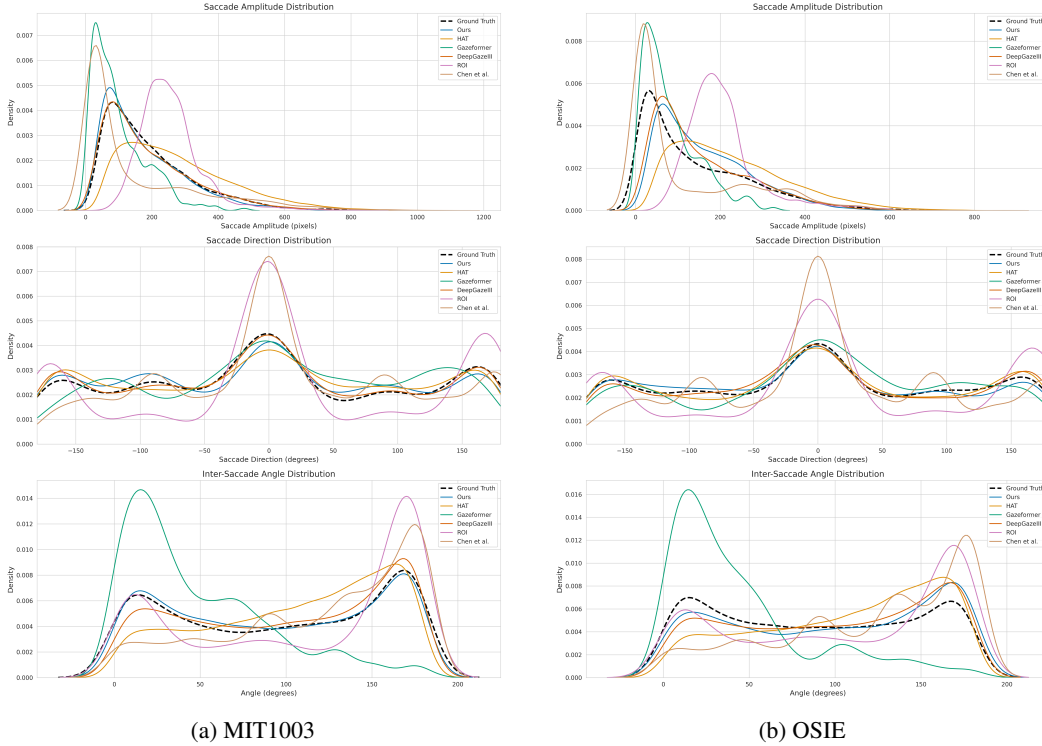


Figure 9: Comparison of statistical properties for generated scanpaths on the (a) MIT1003 and (b) OSIE datasets. The distributions for saccade amplitude, saccade direction, and inter-saccade angle of our model (blue) are compared against ground truth human scanpaths (black, dashed) and several other methods. Our model consistently provides the closest fit to the ground truth distributions.