

MODELING OF EYE CONTACT BEHAVIOR

ZAHRA HOSSEINI

A THESIS SUBMITTED TO THE FACULTY OF GRADUATE STUDIES
IN PARTIAL FULFILMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER OF APPLIED SCIENCE

GRADUATE PROGRAM IN ELECTRICAL ENGINEERING & COMPUTER SCIENCE
YORK UNIVERSITY
TORONTO, ONTARIO

DECEMBER 2024

© AUTHOR, 2024

Abstract

With the rise of online platforms and avatar-based communication, understanding eye contact a key non-verbal cue is crucial for trust in conversations. This study examines eye contact behavior across face-to-face interactions, a screen-sized window interface, and online meetings.

We collected twelve hours of eye contact data from 48 individuals using eye trackers and motion capture in dyadic settings. Our analysis showed consistent eye contact patterns in face-to-face and screen-sized window interactions, while online meetings caused significant shifts due to the lack of direct eye contact.

To model this behavior, we trained a diffusion model (DDPM) to generate synthetic eye movements that preserved key features of real data. We evaluated our model using metrics such as eye contact frequency. This study provides insights into how communication media influence gaze behavior and explores methods for generating realistic eye movements in conversational settings.

Dedication

In honor of Sadaf Hajiaghavand, Pegah Safar Poor Kolor, and Masoud Shaterpour Khiaban, York University students, who lost their lives in the tragic crash of Ukrainian International Airlines Flight 752 near Tehran on January 8, 2020. May their memories live on.

Acknowledgements

I would like to express my deep gratitude to my supervisors, Dr. Nikolaus Troje and Dr. Konstantinos Derpanis for their guidance and support throughout these years. I truly enjoyed this journey and gained numerous skills and memories that I will cherish.

I extend my heartfelt thanks to my parents, who have supported me unconditionally despite our 28-month separation. Their daily love and encouragement have been a constant source of motivation, and I owe everything I am to them.

I am especially thankful to my sister, Mahsa, who inspired me from an early age by showing me the beauty of mathematics and sparking my interest in research later. I am also grateful to my brother, Mohammadreza, whose presence has always brought joy and warmth especially during this past year.

A heartfelt thank you goes to my friends, Kimia, Jason, Heeva, Bitu, Pooya. Knowing I had your support was always a source of peace, and I could rely on you whenever I needed someone. You are truly the best.

Lastly, I extend my gratitude to my colleagues at the CVIL Lab and the BioMotion Lab. I am truly fortunate to have been part of such a supportive and inspiring environment, where I could work, learn, and build friendships. Your presence made me feel a sense of belonging in a new country, and for that, I am deeply grateful.

Table of Contents

Abstract	ii
Dedication	iii
Acknowledgements	iv
Table of Contents	v
List of Tables	viii
List of Figures	ix
Abbreviations	xi
1 Introduction	1
1.1 Motivation	1
1.2 Contributions	4
2 Literature Review	6
2.1 Eye Contact Behaviour	6
2.2 Eye Contact Datasets	8
2.3 Behavior Modeling	10
2.3.1 Multi-agent Behavior Modeling	10
2.3.2 Eye Movement Modeling	11
2.4 Generative Models	12
2.4.1 Diffusion Models	12
2.5 Deep Learning Gaze Tasks	14
2.6 Hardware System	15

2.6.1	Motion Capture System	16
2.6.2	Eye Trackers	18
2.7	Summary	20
3	Data Collection	21
3.1	Data Collection Scenario	22
3.2	Hardware Setup	22
3.3	Experimental setup	26
3.4	Room Coordinate Mapping	29
3.5	Mutual Gaze Detection	30
3.6	Data Collection Statistics	34
3.7	Summary	34
4	Data Analysis	35
4.1	Methodology of Data Analysis	35
4.1.1	Data Integration	35
4.1.2	Identification of Eye Contact Events	36
4.2	Data Analysis Results	36
4.2.1	Effect of Condition	36
4.2.2	Effect of Role	40
4.3	Summary	40
5	Modeling	42
5.1	Dyadic Model Properties	42
5.2	Data Representation and Notation	44
5.3	Modeling Eye-Contact	44
5.3.1	Adapting Diffusion Models for Eye-Contact Prediction	45
5.3.2	Handling Constraints in Data Representation	47
5.3.3	Training Procedure	47
5.4	Summary	48
6	Results	49
6.1	Eye Movement Event Detection	49
6.2	Quantitative Results	52
6.2.1	Single Eye Analysis	52

6.2.2	Interocular Eye Coordination (Vergence)	53
6.2.3	Eye-Head Coordination	55
6.2.4	Dyadic Coordination	57
6.3	Qualitative Results	57
6.4	Summary	59
7	Discussion	60
7.1	Achievements	60
7.1.1	Novel Dataset for Dyadic Gaze and Head Movement	60
7.1.2	Realistic Generative Model	61
7.1.3	Validation through Metrics and Visualization	61
7.2	Limitations	61
7.2.1	Simplified Interaction Scenarios	61
7.2.2	Lack of Contextual Awareness	61
7.2.3	Overestimated Eye Contact Frequency	62
7.3	Future Work	62
7.4	Societal Impact	63
7.5	Conclusion	64
	Bibliography	64
A	Appendix	74
A.1	Demographics of Participants	74
A.1.1	Age Distribution	74
A.1.2	Gender Distribution	76
A.1.3	Visual Acuity and Prescription Needs	76
A.2	Visualization Tool for Collected Data	77
A.3	Calibration of Zoom Camera and Screen Mapping	78

List of Tables

2.1	Summary of eye Contact Datasets	10
3.1	Eye Tracker's Data Columns	29
3.2	Experiment Statistics	34
6.1	Comparison of Single Eye Metrics for Generated and Real Data	53
6.2	Head and Gaze Correlation Values in Real and Generated Data	56
6.3	Comparison of Eye Contact Metrics for Real and Generated Data	57

List of Figures

1.1	Tobii Pro 3 Wearable Glasses	3
1.2	OptiTrack Motion Capture System and Camera	3
2.1	Training and Sampling in Diffusion Models	13
2.2	Hardware Setup	16
2.3	Media Coordinate System in Tobii Eye Trackers	19
3.1	Tobii’s Marker Sets	23
3.2	Data Collection Process Overview	24
3.3	TTL Pulse Pattern	25
3.4	Face-to-Face Interaction	27
3.5	Through-a-Window	27
3.6	Over an online meeting platform	28
3.7	Eye Contact Detection Process	32
4.1	Direct Gaze Duration and Direct Gaze Frequency	37
4.2	Simultaneous Direct Gaze Frequency and Simultaneous Direct Gaze Duration	38
4.3	Direct Gaze Probability	39
4.4	Probability Analysis of Gaze Events	39
4.5	Role Effect on Gaze Behaviour	40
5.1	Training Process Overview	46
6.1	Eye Movement Types Detection	51
6.2	Comparison of Real and Generated Gaze Difference Distributions	54
6.3	Yaw and Pitch Axes of Head and Gaze	55
6.4	Ready Player Me Avatars	58

A.1	Age Distribution	75
A.2	Gender Distribution	76
A.3	Visual Acuity Distribution	77
A.4	Visualization Tool	78
A.5	Zoom Camera Calibration	79

Abbreviations

- 3D: Three-Dimensional
- AI: Artificial Intelligence
- ANOVA: Analysis of Variance
- API: Application Programming Interface
- CNN: Convolutional Neural Network
- CSV: Comma-Separated Values
- DDPM: Denoising Diffusion Probabilistic Model
- FPS: Frames Per Second
- GAN: Generative Adversarial Network
- GPU: Graphics Processing Unit
- HCI: Human-Computer Interaction
- NI-DAQ: National Instruments Data Acquisition
- PCA: Principal Component Analysis
- SLERP: Spherical Linear Interpolation
- TTL: Transistor-Transistor Logic
- VAE: Variational Autoencoder

Chapter 1

Introduction

1.1 Motivation

Eye contact plays a fundamental role in face-to-face communication, conveying critical information through both conscious and unconscious cues. It facilitates attention, trust, and engagement, enabling individuals to interpret social signals, understand emotions, and coordinate interactions. For example, mutual gaze often signifies interest and attentiveness, while looking away may indicate discomfort or disinterest. This intricate dynamic is essential for effective communication in personal and professional settings [1, 2].

Despite eye contact’s significance in human communication, replicating natural eye contact behavior in artificial systems, such as visual chatbots, remains a considerable challenge. As today, online meetings are integral to our daily lives, and in the near future, interactions with visual chatbots will be part of our daily interactions as well. Therefore, simulating realistic eye contact becomes increasingly important for enhancing social presence and interaction quality. This study tries to deepen our understanding of eye contact behavior and explore whether a machine-learning model can learn and generate such behaviors. By addressing this question, we pave the way for the future development of systems capable of delivering more natural and engaging interactions in virtual environments. Current datasets and models often fail to capture the intricacies of real-world interactions, particularly the dynamic and mutual nature of gaze between two individuals.

Motivated by the need to address these gaps, this work aims to design a system that generates realistic, socially competent eye movement behaviors. Such a system requires modeling dyadic interactions by conditioning predictions on both participants’ immediate past and current behaviors. Our goal is to advance human-computer interaction by enabling

artificial systems, such as visual chatbots and social robots, to engage in natural and context-aware eye contact, fostering more intuitive and human-like communication.

To pursue this goal, we collaborated with psychologists to expand the scope of our knowledge of eye contact behavior, which included recording hours of data that only part of it used for training a generative model. This additional data provided insights into eye contact behavior and variability across different communication scenarios, enriching our understanding of the problem space. However, challenges such as the difficulty of modeling complex social cues and limited data diversity meant that the system fell short of achieving fully autonomous and context-aware eye movement generation. Future work should focus on integrating more advanced contextual models, increasing data richness, and refining generative methods to bridge these gaps and further approach the vision of socially competent artificial systems.

Dataset and Generative Model Motivation The collection of this dataset is highly resource-intensive, both in terms of time and cost. To support future research, our project creates a new dataset that includes the eye movements of two participants engaged in conversation while exploring the use of generative models to simulate similar data. Unlike existing datasets that typically track only one person’s gaze and are often too short for comprehensive learning applications [3, 4], our dataset captures the dynamics of mutual gaze and eye contact between participants over extended periods. By using a generative model, we aim to see if we can produce realistic eye and head movements, potentially reducing the need for extensive data collection in the future. This approach lays the foundation for more efficient research and provides a comprehensive representation of eye contact behavior across various communication settings.

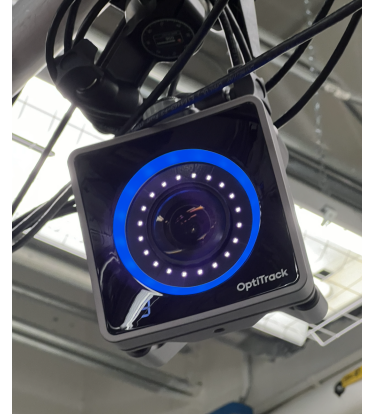
Experimental Setup and Data Collection To achieve this, we designed an experiment to collect synchronized eye and head movement data using advanced eye tracking and motion capture technologies. We used Tobii Pro 3 wearable glasses [5] for eye tracking (see Figure 1.1) and OptiTrack systems [6] for capturing head movements (as shown in Figure 1.2). The Tobii glasses consist of a head unit and a recording unit. The head unit resembles everyday glasses, making it comfortable and natural for participants.



Figure 1.1: Tobii Pro 3 wearable glasses equipped with reflective markers for motion capture integration.



(a) Our OptiTrack motion capture system setup.



(b) A close-up view of an OptiTrack motion capture camera.

Figure 1.2: Illustration of the OptiTrack motion capture system. (a) Shows the overall system setup, including the arrangement of cameras and reflective markers. (b) Depicts a close-up of a motion capture camera, highlighting the technology used to detect movements in 3D space.

We collected data under three different conditions to study eye contact behavior in various scenarios: face-to-face conversation, conversation through a window, and conversation over Zoom. These diverse settings allow us to better understand how eye contact varies across different communication channels. To ensure that participants felt comfortable and to

facilitate the collection of realistic data, they were asked to play the game “Heads Up” during the recordings.

Generative Model Approach After conducting thorough data analysis and evaluating available models, we employed diffusion models to learn and generate head and eye movements, including gaze direction. Diffusion models simulate natural dynamics, and we further enhance them with transformer-based architectures to learn sequence-based features of the data. The transformer component allows the model to handle variations in the order of data in our dyadic setting, making the model more robust and versatile.

Evaluation of Generative Models Evaluating generative models is inherently challenging, requiring a combination of quantitative and qualitative approaches. Quantitatively, we assessed the generated behaviors by analyzing key metrics such as saccade frequency, fixation durations, eye-head coordination, and eye contact patterns. These metrics were compared against real data to validate the model’s ability to replicate realistic social interactions. Qualitatively, we visualized the generated movements using avatars in a Unity-based application, providing an intuitive way to assess the naturalness of the results.

1.2 Contributions

This work introduces several key advancements compared to previous studies. Unlike existing datasets, which primarily rely on images or videos without ground truth eye contact data, our dataset utilizes advanced eye-tracking and motion capture technologies to record synchronized eye and head movements accurately. Our generative modeling approach further enhances this work by leveraging diffusion models to produce eye and head movements in a dyadic setting.

- **Data Collection and Synchronization in Diverse Communication Scenarios:** We collected data under three distinct conditions to analyze how eye contact behaviors vary across different communication mediums.
- **Direct Analysis of Collected Data:** We conducted a thorough analysis of the data collected, examining the dynamics of eye contact and mutual gaze behavior under different communication conditions, which helped us better understand eye contact behavior.

- **Development and Evaluation of a Diffusion Model with Transformer-based Approach:** We employed diffusion models to learn the temporal and sequential features of the collected data. The generated eye contact behaviors were evaluated using both qualitative and quantitative methods.

Chapter 2

Literature Review

2.1 Eye Contact Behaviour

A non-verbal social cue refers to the use of body language, facial expressions, gestures, eye contact, and other physical behaviors that convey messages or emotions without using words [7]. These cues are critical in social interactions as they help regulate communication, express emotions, signal attention and engagement, and facilitate the understanding of social dynamics [8].

In the context of eye contact, it plays a significant role as a non-verbal social cue in human communication. Eye contact helps to make trust during a conversation, and finding a fellow in conversational turn-taking, express emotions, and enhancing the overall quality of interactions. Additionally, studies have shown that variations in eye contact behavior can impact the perception of trustworthiness, engagement, and naturalness during conversations [9, 10].

These cues are essential for creating realistic interactions in both real-world and virtual environments [11], where modeling and simulating eye contact can significantly affect the perceived quality of communication.

Eye contact in real life is a complex and essential component of human social interaction, playing a crucial role in communication, emotional expression, and social bonding. It occurs naturally when individuals look towards each other's faces, creating a shared sense of presence and connection. However, it is important to distinguish between general eye contact and direct gaze. Eye contact broadly involves mutual recognition of each other's gaze, often signaling engagement and attention, while direct gaze specifically refers to looking directly into another person's eyes, which can intensify the interaction by conveying intent, confidence,

or emotional resonance. Real-life eye contact is influenced by various factors such as social context, cultural norms, and the perceived authenticity of the interaction, making it a nuanced aspect of human behavior that extends beyond simple visual alignment.

Holleman [8] investigated how implied social interaction influences gaze behavior, highlighting the nuanced nature of eye contact. In their study, participants viewed a recorded confederate under varying conditions, which affected their belief in the video’s live nature. Factors such as the reactivity of the confederate, environmental consistency, and the overall context influenced how participants perceived the authenticity of the interaction. These findings underscore the complexity of interpreting eye contact in social settings, suggesting that the perception of authenticity plays a significant role in how eye contact is experienced.

In a similar vein, Hietanen and Hietanen [12] explored the impact of direct gaze on self-awareness and self-referential processing. Their experiments revealed that while video-based eye contact did not significantly affect participants, live direct gaze increased self-referential processing, indicating that the psychological impact of direct gaze is increased when the eye contact feels genuine.

The research by Holleman emphasizes that the authenticity of the interaction profoundly affects how eye contact is perceived, showing that environmental and contextual cues can alter the meaning and effectiveness of gaze behavior. Hietanen and Hietanen’s findings further suggest that direct gaze, especially in genuine face-to-face settings, can deeply influence self-perception and social dynamics. Understanding these distinctions is crucial for appreciating the layered nature of human interactions and underscores the challenges of replicating these nuanced behaviors in virtual environments, where the authenticity and emotional impact of direct gaze are often diminished.

Non-verbal social cues especially eye-contact are essential in online meetings, such as those conducted on platforms like Zoom, because they significantly impact the quality and effectiveness of communication. These cues, including eye contact, facial expressions, and gestures, help regulate turn-taking, allowing participants to know when to speak or listen, which is particularly important in virtual settings where slight delays can disrupt conversational flow. Non-verbal behaviors also enhance engagement and connection by showing active participation, such as nodding or maintaining eye contact with the camera, which makes participants feel acknowledged and valued despite the virtual barrier. Non-verbal cues also signal focus and attention, as maintaining eye contact with the camera simulates direct gaze, indicating attentiveness, even though it lacks the natural feedback loop

of in-person eye contact. This simulation of non-verbal communication plays a crucial role in making remote meetings feel more personal and connected.

However, current online meeting platforms have significant limitations in effectively conveying non-verbal cues. Achieving true eye contact in online meetings is challenging due to the misalignment between the camera and the screen. When participants look at their screens, they are often either focusing on the other person’s video feed, their own picture, or other elements on the screen, which gives the impression that they are looking away from the camera [13]. In the best-case scenario, participants might look directly at the camera to simulate direct gaze; however, this still does not achieve real eye contact because it lacks the mutual visual feedback that occurs in face-to-face interactions. This misalignment disrupts the natural dynamics of gaze behavior, where eye contact is crucial for signaling attention, engagement, and understanding, ultimately diminishing the effectiveness of non-verbal communication in virtual meetings.

Currently, VR and online meeting platforms use advanced technologies like NVIDIA Maxine [14, 15] and Apple ARKit [16] to enhance virtual communication. NVIDIA Maxine offers AI-powered features such as noise cancellation, virtual backgrounds, and simulated eye contact, which aligns users’ gaze with the camera to improve engagement. Apple ARKit provides real-time facial tracking, enabling realistic avatar movements and expressive virtual interactions. Despite these advancements, these technologies still fall short of fully replicating the nuanced non-verbal cues of in-person communication, such as true motion parallax, subtle gaze dynamics, and natural depth perception, making virtual interactions feel less immersive and authentic.

2.2 Eye Contact Datasets

Eye contact detection is a critical area of research with significant implications for applications in visual chatbots, social robots, and human-computer interaction. Despite its importance, the availability of datasets designed explicitly for eye contact detection has been limited. The early datasets in this field were not primarily focused on eye contact; instead, they were adapted from other domains, such as facial analysis and face recognition, using gaze detection algorithms to label eye contact instances indirectly. This lack of dedicated, ground-truth datasets has presented challenges in developing robust models for eye contact detection.

The DEEPEC (DEEP Eye Contact Detection) dataset, developed by Mitsuzumi et al. [17], represents one of the first significant efforts in this space. DEEPEC utilizes a two-stream, six-layer convolutional neural network (CNN) architecture to detect eye contact from images of the left and right eyes. The dataset, which includes 4,150 manually annotated samples sourced from LFPW [18], Helen [19], AFW [20], and IBUG [21], provides a balanced representation of eye contact and averted gaze. However, its primary limitation is its focus on high-resolution images intended initially for facial analysis. This restricts its application in more dynamic and real-world scenarios where eye contact plays a crucial role.

Another dataset, OFDIW (Onfocus Detection in the Wild) [22], extends the eye contact detection capability to both humans and animals. Zhang et al. [22] created this dataset using images from the LFW (Labeled Faces in the Wild) collection [18], initially designed for face recognition tasks. The human dataset comprises 16,548 images and adds valuable data for training models in more varied conditions. However, similar to DEEPEC, the images in OFDIW were not collected explicitly for eye contact detection, which may impact the precision of its annotations for this specific task.

A more substantial but less accessible contribution comes from Chong et al. [23], who developed a large-scale dataset for their study on eye contact detection with deep neural networks. This dataset includes over 4 million annotated frames, with a significant portion labeled as eye contact. Despite its size and potential utility, this dataset is not publicly available, limiting its impact on the broader research community and its application in advancing eye contact detection technologies.

In contrast to these works, our dataset addresses these limitations by providing a ground truth eye-contact dataset collected from dynamic, naturalistic interactions rather than relying on static high-resolution images or gaze estimation tasks applied to videos or images. Specifically, we utilized two eye trackers to capture the gaze behavior of both participants in a dyadic interaction and synchronized, recording sequences of their gaze locations throughout the conversation. This setup allowed us to accurately label eye contact events based on precise gaze tracking data, eliminating the need for post-hoc labeling or indirect estimation methods.

Table 2.1: Compact summary of datasets for eye contact detection, including data type, labeling methods, size, and labeling approaches.

Dataset	Data Type	Labeling Method	Amount	Method/Model
DEEPEC [17]	Eye images	Manual	4,150	Two-stream CNN
OFDIW [22]	Human/animal images	Gaze detection	16,548	LFW-derived labels
Chong et al. [23]	Video frames	Semi-auto (DNN)	4M+	Deep neural network
Our Dataset	Dyadic gaze behavior	Precise tracking	11.2 hrs	Direct gaze tracking

In summary, earlier datasets such as DEEPEC and OFDIW (Table 2.1) laid the groundwork for eye contact detection research. Still, they faced limitations due to their indirect labeling methods and focus on static images. The large-scale dataset by Chong et al. offered significant volume but remains inaccessible to the broader community and still needs gaze estimation on video frames. Our dataset addresses these gaps by providing high-precision, ground-truth annotations based on naturalistic dyadic interactions, enabling the development of robust and context-aware models for eye contact detection.

2.3 Behavior Modeling

Behavior modeling has gained significant attention as researchers strive to understand and forecast human behavior in various contexts. This work spans a broad spectrum, from trajectory prediction and forecasting in physical spaces [24] to more complex behavioral patterns such as facial expression generation [25]. The overarching goal is to create models that can accurately predict human actions and reactions, which is crucial for applications in human-computer interaction, robotics, and social signal processing. This section explores key areas within behavior modeling, specifically focusing on dyadic behavior modeling and eye movement modeling, which are essential for understanding interactions between individuals.

2.3.1 Multi-agent Behavior Modeling

Dyadic behavior modeling focuses on learning and predicting interactions between two agents, such as in conversations, negotiations, or collaborative tasks. Dyadic interactions are inherently complex as they involve continuous exchanges of verbal and non-verbal signals, including gestures, gaze, and facial expressions. Müller et al. [26] provided a contribution to this field by modeling dyadic interactions using video data from YouTube interviews. They

employed a recurrent neural network with Long Short-Term Memory (LSTM) units [27] to predict gaze aversion, a critical non-verbal cue in conversations. Their approach incorporated multimodal features such as head pose, gaze direction, and facial expressions, showing results in classifying eye contact and averted gaze.

Recent studies have extended dyadic behavior modeling to include the prediction of backchannel responses and other social signals [28]. These multimodal combinations have proven effective in predicting both the occurrence and subtype of backchannels, contributing to more engaging and responsive human-agent interactions.

Interaction modeling focuses on predicting and understanding the dynamics of social and physical interactions between agents, often in crowded or multi-agent scenarios. This field builds on trajectory prediction and extends to more complex multi-modal interactions. Several approaches have been developed to model these behaviors effectively.

Alahi et al. [29] introduced Social LSTM, which employs a recurrent neural network architecture to predict the trajectories of multiple interacting agents. By using a social pooling mechanism, this model captures the influence of neighboring agents on an individual’s trajectory, demonstrating significant improvements in accuracy for crowd navigation tasks.

Gupta et al. [30] extended this work with Social GAN, a generative adversarial network-based approach that generates socially acceptable trajectories. By incorporating stochasticity, Social GAN captures the multi-modal nature of human interactions, offering diverse trajectory predictions.

More recently, Yuan et al. [31] proposed Agentformer, a transformer-based framework for modeling multi-agent trajectories. By leveraging attention mechanisms, Agentformer effectively captures the interdependencies among agents, outperforming previous methods in trajectory prediction tasks.

2.3.2 Eye Movement Modeling

Eye movement modeling is vital for understanding how people engage with their environment and others, especially in conversational contexts. Despite its importance, there is limited research specifically targeting the modeling of eye movements during interactions. The tasks typically involve gaze estimation, eye contact detection, and gaze-following, which are crucial for predicting attention and engagement in social settings. Studies such as Huang et al. [32] and Tesema et al. [33] highlight the significance of multi-modal approaches, integrating verbal and visual features to predict nodding and attention-switching behaviors in human-robot

interactions. However, much of the existing work focuses on head movement and turn-taking, often using eye movements as secondary features [3].

Jiang et al. [34] introduced MotionDiffuser, a diffusion-based generative model for learning complex motion patterns in multi-agent settings. This approach provided insights into leveraging diffusion models to model behaviors such as eye contact and attention dynamics, inspiring advancements in generative modeling for social interactions. Motiondiffuser’s ability to model fine-grained interaction details aligns with the challenges of learning eye contact as a multi-agent behavior.

These approaches collectively highlight the potential of leveraging advanced generative and sequence modeling techniques for interaction modeling. The use of generative models, specifically diffusion models, offers promising directions for capturing multi-agent behaviors, including eye contact, as dynamic and context-sensitive phenomena.

2.4 Generative Models

Generative models aim to learn the underlying probability distribution of a dataset, enabling the generation of new data points that are statistically similar to the observed data [35], and the evaluation of likelihoods for new unseen data. These models have been widely applied in various domains, including image generation [36], video generation [37], and even temporal data modeling [38], where capturing the complexity of the data distribution is the critical. Various formulations have been proposed that offer different tradeoffs in sample quality, computational costs when sampling, and their ability to provide accurate likelihoods. Methods such as PixelCNNs [39] and normalizing flows [40] directly model a likelihood function, while VAEs [41] seek to maximize a lower bound on a likelihood function. Other methods such as GANs [35] indirectly model the distribution without likelihoods, and are instead optimize by an adversarial objective. In this thesis, we are most interested in exploring the use of diffusion models [42], a recent method capable of producing high quality samples. These models are based on learning the reverse of a fixed noising process, which has been shown to represent the score of a likelihood function [43, 44].

2.4.1 Diffusion Models

Diffusion models operate on a principle that contrasts with typical generative approaches. Unlike methods that directly model the data distribution, diffusion models [45, 46] define a

fixed forward process that gradually adds noise to the data over a range of time steps $t \in T$, effectively transforming it into a normal distribution.

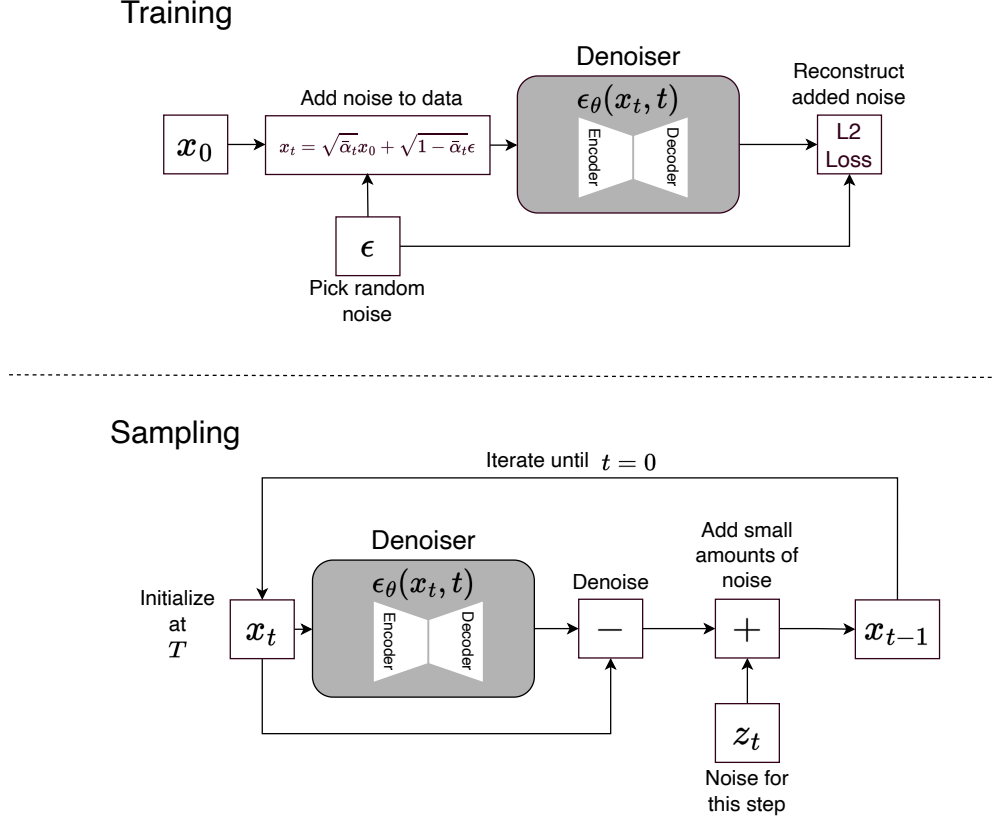


Figure 2.1: Illustration of the training (top) and sampling (bottom) processes in diffusion models. Training involves adding noise to data and reconstructing it using a denoiser. Sampling iteratively removes noise to generate data.

This process is formally defined as: (See Figure 2.1, top panel, for an illustration of this process.)

$$x_t = \sqrt{1 - \beta_t}x_{t-1} + \beta_t\epsilon, \quad \epsilon \sim \mathcal{N}(0, I), \quad (2.1)$$

where x_t is the noised data at time t , β_t are coefficients defined by fixed noise schedule, and ϵ is sample of Gaussian noise.

The reverse process, which allows the model to draw samples, involves learning the reverse of the forward diffusion process, transforming a sample of normally distributed noise back into structured data. This reverse diffusion effectively acts as a “de-noising” process that reconstructs the original data distribution by gradually transforming the original noise sample

to sample close to the desired data distribution. Specifically, the reverse process is also a Gaussian process:

$$x_{t-1} = \mu_{\theta}(x_t, t) + \Sigma_{\theta}(x_t, t)\epsilon, \quad \epsilon \sim \mathcal{N}(0, I), \quad (2.2)$$

where, the mean μ and standard deviation Σ of the process is approximated by a neural network with parameters θ . In essence, the reverse process is trained to predict earlier, less-noisy data points x_{t-1} from more-noisy data points x_t .

The training objective for this neural network is to match the predicted distribution of x_{t-1} , conditioned on x_t . When simplified, the loss function conventionally used is:

$$\mathcal{L} = \mathbb{E}_{x_0, t, \epsilon} [||\epsilon - \epsilon_{\theta}(\sqrt{\bar{\alpha}_t}x_0 + \sqrt{1 - \bar{\alpha}_t}\epsilon, t)||^2], \quad (2.3)$$

where for a randomly selected training example x_0 , time t , and noise ϵ , the neural network is reparameterized using $\epsilon_{\theta}(\cdot)$ to predict the noise ϵ from the noisy data $\sqrt{\bar{\alpha}_t}x_0 + \sqrt{1 - \bar{\alpha}_t}\epsilon$ by minimizing an L2 reconstruction loss. Training the neural network is performed using modern stochastic gradient descent techniques [47].

During sampling, the predicted noise from $\epsilon_{\theta}(\cdot)$ is subtracted from each sample of x_t to get x_{t-1} . This step-by-step denoising process contrasts with other generative models like GANs, which typically involve a single-stage transformation.

The gradual nature of diffusion models allows them to capture complex data distributions more effectively, particularly in high-dimensional spaces.

In the context of our project, x_t contain values for a sequence of gaze directions and head pose. The diffusion process gradually refines a sequence of random head and gaze values into a plausible sequence where the motions should represent continuous motion, capturing both spatial and temporal aspects of gaze behavior.

2.5 Deep Learning Gaze Tasks

Gaze-related tasks, they focus on predicting and understanding where and how individuals direct their attention in various contexts, such as human-computer interaction, or estimating gaze in the world coordinate system.

Mondal et al. introduced Gazeformer [48], a transformer-based model designed to predict human gaze in goal-directed tasks, such as visual search, without requiring extensive labeled training data for every possible target object [48]. Their approach, termed ZeroGaze, enables

scalable gaze prediction by utilizing language models to encode search targets, thereby outperforming traditional gaze models in both accuracy and speed.

Tu et al. proposed HGTTR [49], a transformer-based framework for Human-Gaze-Target detection that integrates the task of identifying human gaze directions and their gaze targets in a unified, end-to-end process [49]. Unlike conventional methods that require separate branches for head and scene understanding, HGTTR simplifies the pipeline by directly predicting gaze targets from the global image context, achieving state-of-the-art performance on standard gaze-following benchmarks.

Pan et al. developed an interactive gaze system for humanoid robots that integrates lifelike gaze behaviors into human-robot interactions [50]. Their system employs an architecture inspired by animation techniques, allowing robots to respond dynamically to human attention cues, thereby enhancing the realism and engagement in human-robot interactions.

These works underscore the growing capabilities of gaze-related modeling and tasks in enhancing interactive systems, with applications spanning from human-computer interaction to robotics.

2.6 Hardware System

Given the absence of dedicated datasets specifically designed to capture eye contact behavior, We developed a ground truth dataset using a motion capture system with markers and eye trackers. The setup employed OptiTrack’s PrimeX 41 [51] cameras. In Figure 2.2, we show the camera setup. Below, we detail the components and workflow of our motion capture system and eye trackers, which integrates advanced technology to provide reliable, high-resolution tracking data.

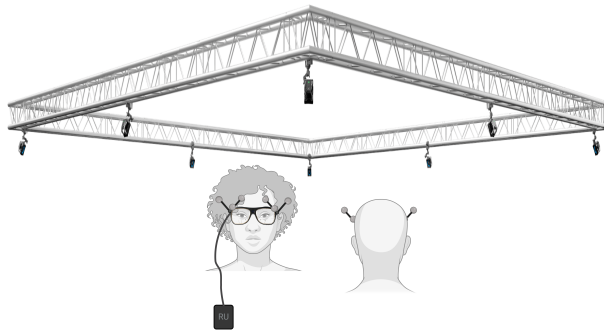


Figure 2.2: Illustration of the motion capture system and eye tracker setup used in the study. Participants were equipped with wearable eye trackers, adjusted to their individual prescriptions to ensure accurate gaze tracking.

2.6.1 Motion Capture System

Our motion capture system utilized the OptiTrack PrimeX 41 [51] cameras, configured in a 14-camera setup, to ensure comprehensive coverage of the capture space. This setup was ideal for capturing detailed head movements required for collecting a ground truth eye contact dataset. A typical motion capture pipeline includes several key stages: calibration, capture session, 2D marker identification, 3D reconstruction, clean up. Each stage is briefly described below:

Calibration: Calibration establishes a unified 3D reference frame for the cameras relative to each other and to the physical capture space. This involves setting up an L-frame with attached markers at the center of the capture space for defining the coordinate system, and defining the origin and orientation of the world coordinate system. Using a Perspective-3-Point (P3P) algorithm, the cameras are calibrated by recording the motion of a wand with attached markers, covering the capture area to create a spatial reference that aligns all cameras to a single coordinate system. The calibration process includes evaluating errors, such as Mean Ray Error, indicating the precision of 3D point calculations, and Mean Wand Error, reflecting discrepancies in detected versus expected wand lengths [52].

Rigid Body: A Rigid Body in motion capture systems is created by securely attaching a set of three or more markers to an unmalleable object, ensuring the distances between markers remain constant during motion. In Motive, this arrangement allows the system to track the object in 6 Degrees of Freedom (6DoF), including position and orientation data

(pitch, roll, and yaw). To define a Rigid Body, markers are placed asymmetrically on the object to avoid identification issues and to provide a clear distinction of orientation [53]. After selecting the markers in the 3D viewport, the Rigid Body is created and labeled, and its pivot point the geometric center is automatically set based on the spatial relationship among the markers. This center point serves as the reference for tracking the Rigid Body’s position and orientation throughout the capture process, ensuring precise monitoring of the object’s movement.

Capture Session: During recording, reflective markers are attached to the subject’s body and face according to a predefined layout that includes tracking of head movements and eye direction. The calibrated cameras track these markers as the subject performs specific tasks or follows given instructions. To minimize artifacts, subjects wear tight-fitting clothing, or markers are placed directly on the skin. The OptiTrack PrimeX 41 cameras capture these motions with high precision, ensuring accurate 3D tracking data [54, 55].

2D Marker Identification and 3D Reconstruction: The system identifies the 2D positions of markers within each camera’s field of view, using infrared LEDs and band-pass filters to enhance marker visibility. Using triangulation techniques, the 3D positions of markers are reconstructed from the 2D data collected by multiple cameras. This results in a set of unlabeled 3D trajectories representing the motion of individual markers over time, providing the raw data needed for further processing [56].

Gap Filling: Gaps in the motion data, often caused by occlusions or noise, are filled using interpolation techniques. While short gaps can be filled accurately with in-built software functions, longer gaps require careful handling as traditional methods can introduce inaccuracies [57, 58, 59].

Denoising: Motion capture data can include noise from various sources, including ghost markers and jitter. Denoising techniques such as low-pass filtering or neural network-based methods are used to clean the data, ensuring smooth and accurate trajectory reconstruction [57, 60].

Overall, the OptiTrack motion capture system provides a highly accurate and reliable method for capturing detailed head, and body movements. Its precision and versatility make it an ideal tool for creating a comprehensive ground truth dataset.

2.6.2 Eye Trackers

Tobii eye trackers, including wearable devices like the Tobii Pro Glasses 3 [61], function by utilizing a combination of advanced hardware components and sophisticated algorithms to accurately capture and analyze eye movements. Below, we outline the main steps involved in how Tobii eye trackers work and the process of collecting eye tracking data.

Setup and Calibration: Before data collection begins, Tobii eye trackers require setup and calibration to ensure accurate gaze tracking. The glasses are fitted to the participant, and calibration is performed using a calibration target. During this process, the participant is asked to look at specific points while the eye tracker records the reflection of near-infrared light patterns on the eyes. This calibration adjusts the eye tracker to the individual’s unique eye characteristics, enhancing the accuracy of gaze data collection [62, 61]. Calibration can be done with a target for optimal precision or automatically for more convenience, although this may reduce accuracy.

Data Capture: Once calibrated, the eye tracker is ready for data capture. Tobii Pro Glasses 3 uses a scene camera mounted at the center of the glasses to record what the participant sees in real time [5]. At the same time, multiple illuminators create a pattern of near-infrared light on the eyes, which is invisible to the user but detectable by the system. Eye cameras positioned within the glasses capture high-resolution images of the user’s eyes and the reflected light patterns, even under varying lighting conditions.

Gaze Data Processing: The captured images are processed by sophisticated image processing algorithms that detect specific details of the user’s eyes, such as the pupil position and corneal reflections [63, 5]. The algorithms use these details to create a 3D eye model, which calculates the eye’s position and orientation relative to the head. By analyzing the vergence of the gaze vectors from both eyes, the system determines the precise gaze point, identifying where the user is looking at any given moment.

Gaze Mapping and Output: The eye tracker projects the calculated 3D gaze point onto the scene camera’s video feed, as shown in Figure 2.3, transforming it into a 2D gaze point relative to the video. This mapping allows researchers to visualize exactly where participants are looking within their environment, offering valuable insights into their attention and behavior [63]. The gaze data is typically exported in formats compatible with Tobii’s software solutions, such as Tobii Pro Lab, where researchers can further analyze and interpret the collected data.

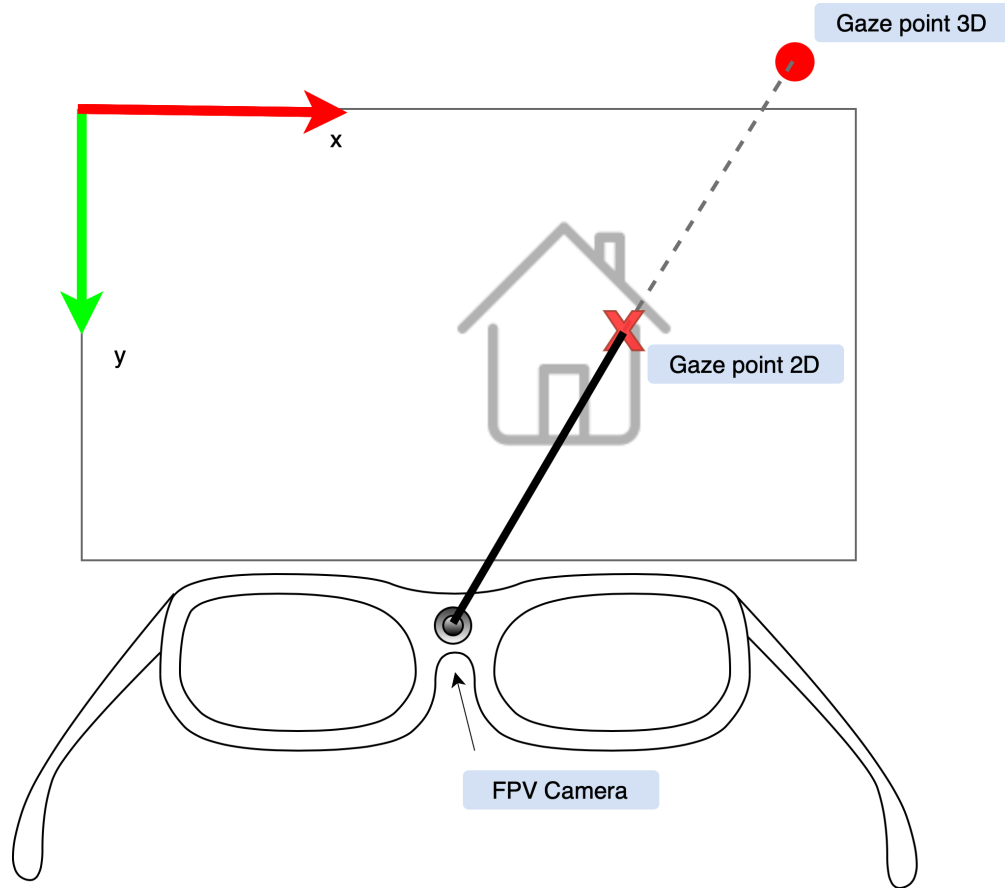


Figure 2.3: Media Coordinate System (MCS) as captured by the Tobii scene camera. The MCS projects the 3D gaze point onto the video feed from the scene camera (first-person view), aligning it within a 2D coordinate system that represents where the user is looking within the captured field of view [63].

Data Analysis: The final step involves analyzing the collected eye tracking data using dedicated software like Tobii Pro Lab or Glasses Explore. Researchers can view the participant's gaze patterns in real-time or replay recorded sessions to explore specific details. These platforms provide advanced analysis tools, such as heatmaps, fixation maps, and statistical metrics, which help in understanding visual behavior, attention, and decision-making processes [62].

Tobii eye trackers provide a comprehensive and precise method for capturing eye movement data across various environments. By combining hardware components like illuminators, eye cameras, and advanced 3D modeling algorithms, Tobii eye trackers enable researchers to collect high-quality gaze data that reflects real-world behavior. This data collection process

is crucial for applications in fields ranging from psychological research to user experience testing, offering deep insights into how individuals interact with their surroundings.

2.7 Summary

This background section reviews key literature relevant to eye contact behavior, data collection, and generative modeling techniques. Section 2.1 explores eye contact as a non-verbal social cue in real-life and virtual settings, highlighting its role in communication and engagement. Section 2.2 discusses the lack of comprehensive datasets capturing natural eye contact in dyadic interactions, emphasizing the need for new data resources. Section 2.3 focuses on behavior modeling, particularly dyadic and eye movement modeling, noting the current emphasis on head and turn-taking movements. Section 2.4 is about generative models, particularly diffusion models, explaining their use in realistic data synthesis. Section 2.5 outlines various gaze-related tasks which is bunch of research work has been done on them. Section 2.6 covers the hardware systems used, including motion capture setups and eye trackers, detailing their methods and precision in tracking gaze and head movements for our data.

Chapter 3

Data Collection

In this chapter, we detail the complete process of collecting 3D eye contact data from human subjects. First, we briefly describe the experimental environment, and the type of data captured from each participant. Then, we detail the specific systems used (i.e., motion capture), and how data from each system is combined to create our dataset. Finally, we also describe how we detect eye contact events from the raw data.

Existing datasets for eye tracking and gaze behavior studies typically focus on capturing data from a single person, limiting the scope of analysis, especially in interactive settings [3]. These one-person datasets fall short in understanding the dynamics of real-time social interactions, such as mutual gaze and eye contact, which are crucial in conversations. Additionally, available datasets are limited in quantity [3], further motivating the need for a richer dataset that includes interactions between two individuals.

Existing eye contact datasets often rely on indirect methods such as gaze estimation algorithms applied to video data, including publicly available sources like YouTube videos [64, 17]. These datasets typically extract eye contact labels using automated or semi-automated processes from general video frames [22], which may introduce inaccuracies and limit their applicability to controlled interaction studies [17]. In contrast, our dataset is specifically designed for high-fidelity, synchronized two-person interaction, capturing detailed 3D head pose and gaze direction data in real-time. This approach provides more precise, ground-truth data that facilitates a comprehensive analysis of eye contact behavior in natural, interactive settings.

To address this gap, we collected a novel two-person dataset that captures head pose and gaze behavior in 3D, focusing on natural, interactive settings between participants.

3.1 Data Collection Scenario

To ensure participants behaved as naturally as possible while wearing eye trackers and interacting with a stranger, we employed a modified version of the “Head’s Up” game; a well-known party game. The modifications were made to adapt the game for use with wearable eye trackers, which required participants to avoid obstructing their foreheads.

In this modified version, participants alternated between two roles: the questioner and the responder. Unlike the original game, where the questioner places a card on their forehead to display their identity, the responder in our study held the card in their hand, and the questioner began asking questions. This adjustment ensured that the eye-trackers remained clear while preserving the game’s interactive nature. The responder was aware of their assigned identity (e.g., a famous person, object, or concept), questioner then attempted to guess the identity by asking a series of yes-or-no questions. The responder, who had access to the questioner’s identity, was restricted to responding only with “Yes” or “No”. If the responder successfully guessed the identity, the roles continued with a new identity card. Each game session lasted eight minutes, with the roles switching midway to ensure balanced participation.

3.2 Hardware Setup

This section will explain the hardware used for capturing 3D head pose and gaze data, specifically the OptiTrack motion capture system and Tobii Pro Glasses 3. We also discuss the synchronization of these two systems to create a unified dataset, which is crucial for accurate analysis of head and gaze behavior during two-person interactions.

OptiTrack for Head Pose Tracking The *OptiTrack* motion capture system was used to track participants’ head movements in 3D space. This system relies on a series of infrared cameras that detect reflective markers attached to the participants’ Tobii glasses. By monitoring the movement of these markers, which are shown in Figure 3.1, the OptiTrack system can compute the 3D position and orientation of the head (i.e., its translation and rotation) with high accuracy. We used two different marker sets to track both participants wearing Tobii Pro Glasses simultaneously in the same motion capture environment. Each marker set has a unique geometric configuration, allowing the motion capture system to distinguish between the two sets of glasses [65].



(a) Tobii marker set 1



(b) Tobii marker set 2

Figure 3.1: Illustration of the unique marker sets attached to Tobii Pro Glasses for seamless integration with motion capture systems. These markers are designed to enable precise tracking of head movements while simultaneously capturing eye gaze data [65].

We need to know where each participant is looking in the room to identify mutual gaze or eye contact events. OptiTrack uses a global coordinate system, which helps us track the head movements of both participants in relation to the room. This setup is essential for understanding how head movements relate to gaze direction and for detecting when participants look at each other.

Tobii Pro Glasses 3 for 3D Gaze Tracking In addition to the head pose, it was critical to capture the eye gaze of each participant. For this, we used *Tobii Pro Glasses 3*, a wearable eye tracker designed for capturing 3D gaze data relative to the participant’s head. The Tobii glasses record the 3D gaze direction by determining the origin and direction of gaze based on the participant’s eye movements. However, Tobii Pro Glasses 3 operate in a local coordinate system that is fixed to the glasses, meaning the gaze data is relative to the glasses themselves and does not account for the participant’s position in the environment. This poses a challenge because gaze direction needs to be interpreted in the context of the participant’s position in the room.

Combining OptiTrack and Tobii Data We needed to combine these three data sources, two eye trackers and motion capture system, to accurately analyze where participants are looking in the room coordinate system. Specifically, we transform the gaze direction data from Tobii’s local coordinate system to the global coordinate system tracked by OptiTrack.

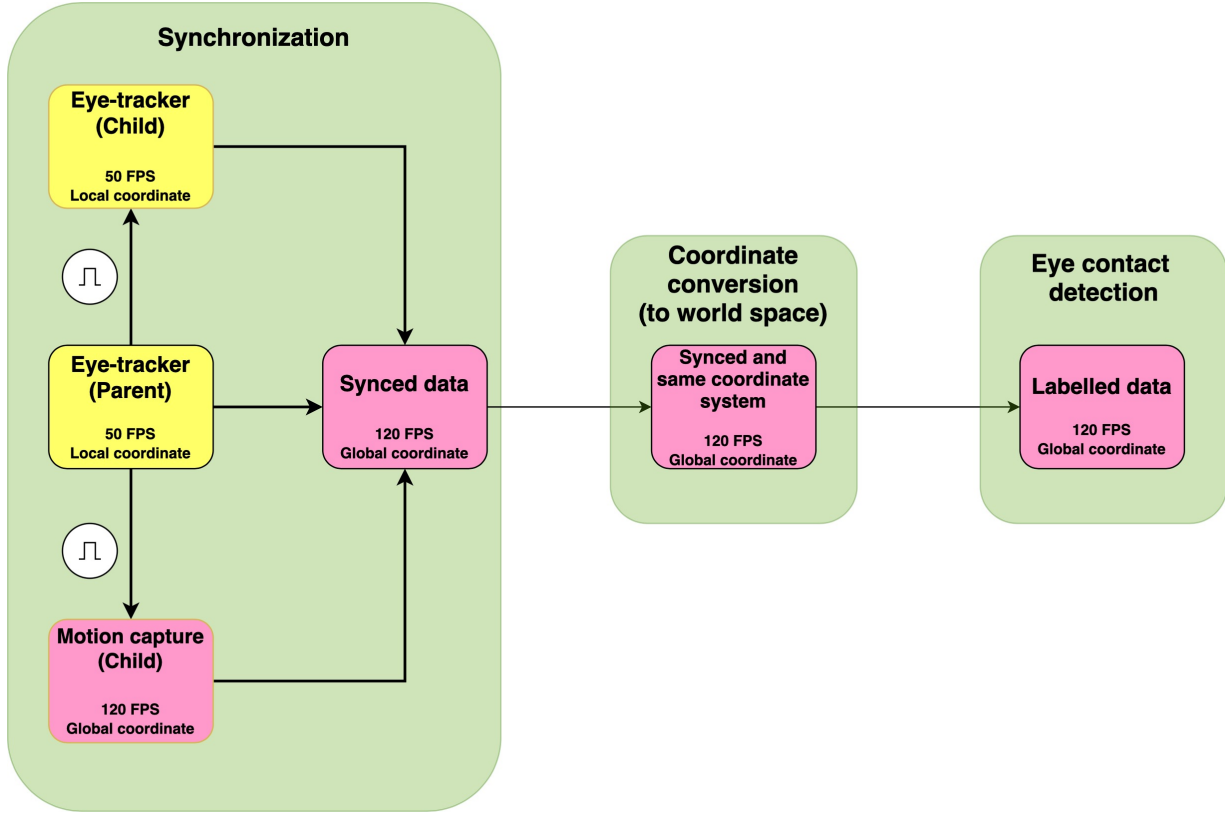


Figure 3.2: Data collection steps: Two eye trackers and a motion capture system are used to synchronize data across different sources. The primary eye tracker sends TTL pulses to the secondary tracker and the motion capture system. Data interpolation techniques are applied to standardize FPS across all data. Next, we converted all the data in the same coordinate system to be able to detect eye contact events; last, we labeled frames where eye contact is happening.

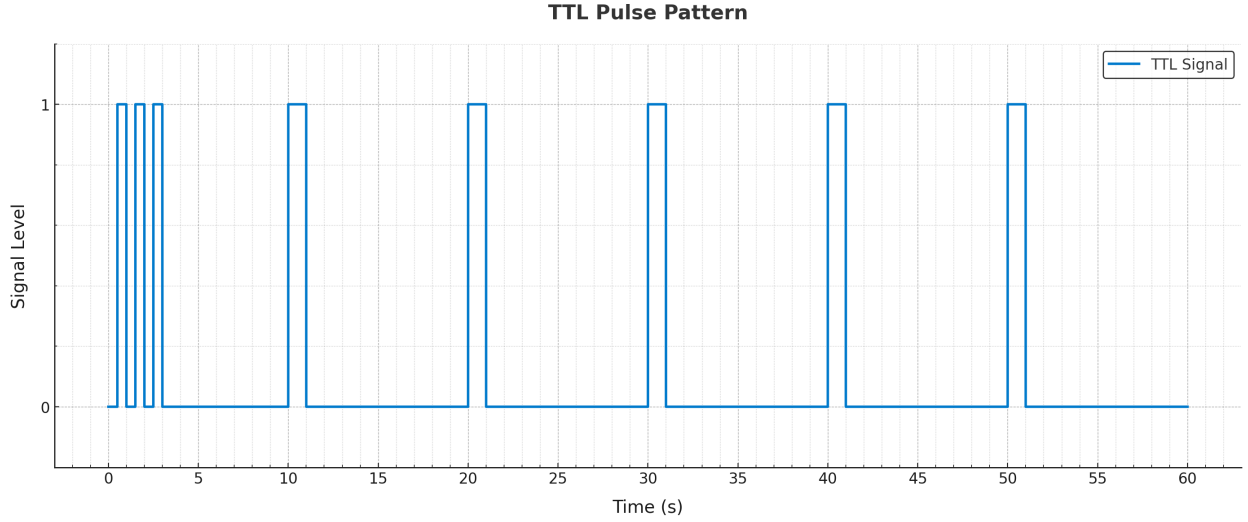


Figure 3.3: TTL pulse pattern, Sequence: When the recording starts, a sequence of 1/0 is sent three times. Each state lasts 500ms. Starting 10 seconds after the first signal, a 1-second pulse is sent every 10 seconds [62].

Synchronization Procedure The synchronization process consisted of the following steps:

1. We attached reflective markers to the Tobii glasses of both participants, enabling the OptiTrack system to track the glasses' position and orientation.
2. An *NI-DAQ* device generated TTL pulses, which were sent to both the Tobii glasses and the OptiTrack system.
3. These TTL pulses were recorded by both systems, providing a common timing reference that allowed us to synchronize the data during post-processing.
4. Each Tobii glasses system was equipped with its own unique marker set, referred to as *Tobii1* ($T1$) and *Tobii2* ($T2$), to differentiate the two participants.

Differences in Capture Rates and Synchronization After synchronization, we addressed the capture rate differences by applying linear interpolation to the lower-frequency Tobii gaze data, which operates at 50 Hz, aligning it with the higher-frequency OptiTrack head pose data, which operates at 120 Hz. For gaze direction vectors, we used spherical linear interpolation (SLERP) to accurately interpolate the 3D vectors. This interpolation allowed us to align head pose and gaze data, ensuring precise detection of mutual gaze and eye contact events between participants.

3.3 Experimental setup

The data capture process involved several key steps to ensure that both head pose and gaze data were collected accurately. Below, we detail the step-by-step procedure followed during each session, starting with system calibration and ending with the data recording for the experimental conditions. Table 3.1 provides an overview of the data columns included in the master dataset, such as timestamps, head pose, gaze coordinates, and pupil diameters for each eye tracker.

Calibration of the Motion Capture System: Before the experiment began, we calibrated the *OptiTrack* motion capture system to ensure accurate tracking of reflective markers attached to the Tobii glasses. This step involved defining the capture volume and verifying that the system was detecting the markers with high precision in the experimental space.

Fitting the Eye Trackers: Each participant was fitted with a pair of *Tobii Pro Glasses 3* to capture their eye gaze. The glasses were adjusted for comfort and proper alignment using adjustable nose pieces, ensuring that the participant’s eyes were positioned in the middle of the tracking area. For participants requiring vision correction, we used the prescription lenses that are coming with the Tobii glasses, fitting them according to the participant’s needs.

Calibration of the Eye Trackers: Once the Tobii glasses were fitted, we calibrated the eye trackers for each participant. The calibration process involved having participants fixate on a specific target positioned in front of them at a recommended distance (typically 50-100 cm). This step adjusted the Tobii system’s gaze estimation algorithms to each participant’s eye characteristics, ensuring accurate 3D gaze tracking during the experiment.

Connecting Participants to the Synchronization System: To synchronize the eye tracking data between the two participants and the motion capture system, we connected each participant’s Tobii glasses via a 3.5mm jack cable. This setup enabled the exchange of TTL pulses between the eye trackers and also linked the system to the *NI-DAQ* device, which generated and sent synchronization signals to the OptiTrack system. This connection ensured that all systems, both eye trackers and the motion capture, were aligned temporally.

Starting the Recording: Once the systems were calibrated and connected, we began data recording. The experiment involved capturing data for three distinct conditions, with participants engaged in a conversation-based activity in each condition:

- **Face-to-Face Interaction:** Participants stand across from each other and engaged in a natural conversation (see Figure 3.4). Both the head pose and gaze data were recorded simultaneously.



Figure 3.4: Face-to-Face Interaction: We asked participants to stand in front of each other without any barriers and start playing the game.

- **Through-a-Window:** A physical barrier with a small window altered the interaction dynamic by restricting their field of view, simulating a partial obstruction during the conversation (see Figure 3.5).

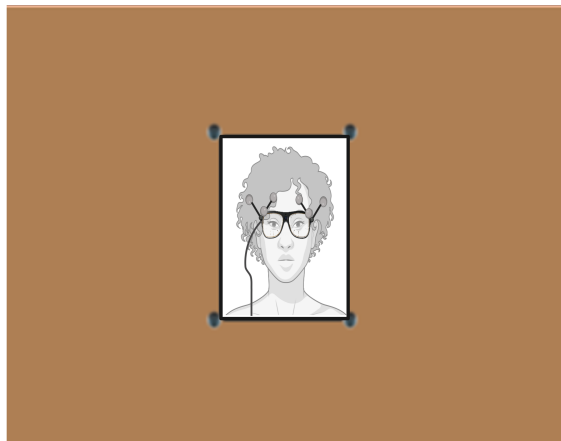


Figure 3.5: Through-a-Window: We put a barrier wall in between participants to study the effect of only the size of an iPad screen on eye-contact behavior.

- **Over an Online Meeting Platform:** Participants used iPads to communicate via an online meeting platform call while still sitting in the same room (see Figure 3.6). This condition allowed us to study the impact of digital communication platforms on gaze behavior.

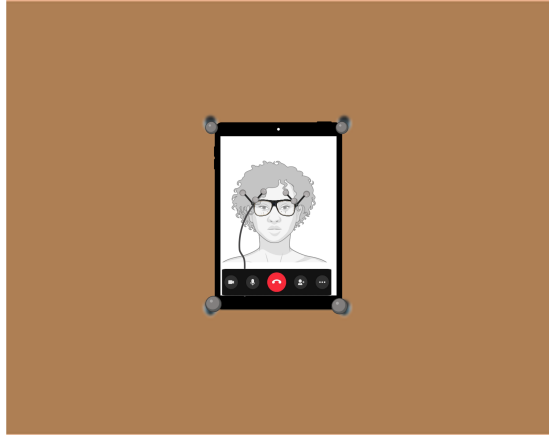


Figure 3.6: Over an online meeting platform: We started an online meeting and asked participants to play the game while they were in an online meeting.

By following these steps, we ensured that the experimental data collection was both accurate and reliable, providing synchronized head pose and gaze data for further analysis. Finally, we have a master dataset for each recoding which includes:

Table 3.1: Explanation of master sheet data columns. T1 refers to eye tracker 1, and the same columns are available for tracker 2.

Column Name	Description
T1.timestamp	Timestamp of the recorded data point
T1.role	Role of the Participant: Questioner/Responder
T1.RX, T1.RY, T1.RZ, T1.RW	Head Rotations (X, Y, Z, W)
T1.TX, T1.TY, T1.TZ	Head Translations (X, Y, Z)
T1.gaze2D.X, T1.gaze2D.Y	2D Gaze coordinates (X, Y)
T1.gaze3D.X, T1.gaze3D.Y, T1.gaze3D.Z	3D Gaze coordinates (X, Y, Z)
T1.leftGazeDirection.X, T1.leftGazeDirection.Y, T1.leftGazeDirection.Z	Left eye gaze direction (X, Y, Z)
T1.rightGazeDirection.X, T1.rightGazeDirection.Y, T1.rightGazeDirection.Z	Right eye gaze direction (X, Y, Z)
T1.leftGazeOrigin.X, T1.leftGazeOrigin.Y, T1.leftGazeOrigin.Z	Origin of the left eye gaze (X, Y, Z)
T1.rightGazeOrigin.X, T1.rightGazeOrigin.Y, T1.rightGazeOrigin.Z	Origin of the right eye gaze (X, Y, Z)
T1.leftPupilDiameter	Diameter of the left pupil
T1.rightPupilDiameter	Diameter of the right pupil

3.4 Room Coordinate Mapping

We convert all gaze data into the same coordinate system to determine where each participant is looking within the room coordinate system. This conversion utilizes rotation and translation values recorded from motion capture system [66]. Given that participants are moving and each frame has different rotation (quaternion) and translation values, a rotation matrix must be created for each data row.

Quaternions are used to represent rotations in three-dimensional space [67, 68]. A quaternion q consists of four elements (q_w, q_x, q_y, q_z) . The corresponding rotation matrix R is derived from the quaternion as follows:

$$R = \begin{pmatrix} 1 - 2(q_y^2 + q_z^2) & 2(q_x q_y - q_z q_w) & 2(q_x q_z + q_y q_w) \\ 2(q_x q_y + q_z q_w) & 1 - 2(q_x^2 + q_z^2) & 2(q_y q_z - q_x q_w) \\ 2(q_x q_z - q_y q_w) & 2(q_y q_z + q_x q_w) & 1 - 2(q_x^2 + q_y^2) \end{pmatrix},$$

and the translation vector T is given by:

$$T = \begin{pmatrix} t_x \\ t_y \\ t_z \end{pmatrix}.$$

To convert the gaze data from the camera coordinate system to the room coordinate system, we apply the rotation matrix R and the translation vector T of head to each data point P :

$$P_{room} = R \cdot P_{camera} + T,$$

Where P_{room} represents the gaze point in the room coordinate system, and P_{camera} represents the gaze point in the camera coordinate system. This transformation ensures that all gaze data points are aligned within the same coordinate system, allowing for accurate analysis of participants' gaze behavior in the context of the room. same formula was applied for gaze origin data.

For gaze direction data, only the rotation matrix R is applied. This is because gaze direction vectors represent orientations rather than positions, and thus do not require translation [69]. The gaze direction vector D_{camera} in the camera coordinate system is transformed to the room coordinate system D_{room} using the rotation matrix R :

$$D_{room} = R \cdot D_{camera},$$

Gaze direction vectors describe the orientation or direction in which a participant is looking, not a specific location in space. Therefore, they are inherently direction vectors that indicate the line of sight from the eye's origin point in a given direction.

3.5 Mutual Gaze Detection

To detect mutual gaze events, we developed a method that transforms the gaze data of each participant into a common global coordinate system, calculates gaze direction vectors, and determines if these vectors intersect with the head plane of the other participant.

Defining the Gaze Direction Vector Rather than using the raw left and right eye data from the Tobii Pro Glasses, which may contain missing values or discrepancies, we calculate

a single gaze direction vector $\mathbf{v}$ for each participant. This vector extends from the midpoint between the eyes to the estimated 3D gaze point ($\mathbf{G}_{3D}$):

$$\mathbf{v} = \mathbf{G}_{3D} - \mathbf{M},$$

where $\mathbf{M}$ is the midpoint between the left and right eye origins:

$$\mathbf{M} = \frac{\mathbf{O}_{left} + \mathbf{O}_{right}}{2},$$

by using the midpoint between the eyes, we eliminate noise and inconsistencies, ensuring a more stable gaze direction. This vector represents the participant's line of sight in 3D space.

Defining the Head Plane To determine if a participant is looking at their partner, we model each participant's head as a plane. This plane is defined using the head pose data from the OptiTrack system, which provides the rotation and translation of the head. The normal vector $\mathbf{n}$ to the plane is derived from the last column of the rotation matrix R :

$$\mathbf{n} = \begin{pmatrix} R_{13} \\ R_{23} \\ R_{33} \end{pmatrix},$$

the equation of the plane is then defined as:

$$n_x x + n_y y + n_z z + d = 0,$$

where d is calculated using the left eye origin $\mathbf{O}_{left}$ as:

$$d = -\mathbf{n} \cdot \mathbf{O}_{left};$$

this plane provides a surface against which we can check whether the gaze vector of the other participant intersects, signaling that they are looking at each other (see Figure 3.7 for an illustration of the eye contact detection process).

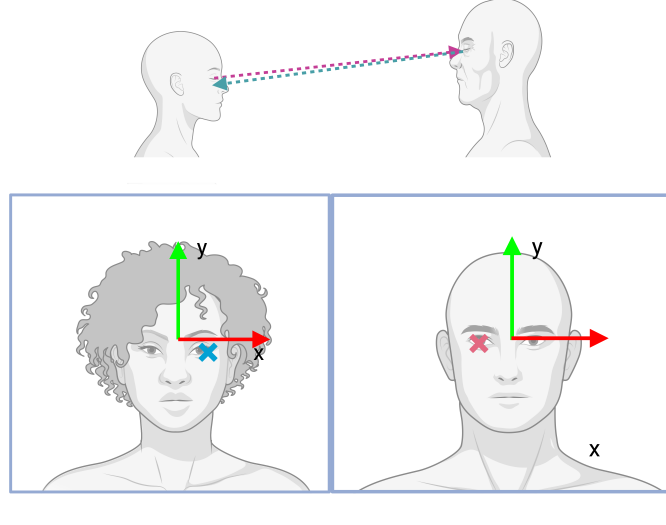


Figure 3.7: Each participant’s gaze direction vector is computed from the midpoint between their eyes to the estimated 3D gaze point. The head plane of the other participant is modeled using head pose data, including rotation and translation, captured by the OptiTrack system. Mutual gaze is detected when the gaze direction vector of one participant intersects the head plane of the other at the same time. This method ensures accurate identification of mutual gaze events in three-dimensional space.

Global to Local Coordinate Transformation Because the gaze data is recorded in a local coordinate system relative to each participant’s head, we must transform it into the global coordinate system tracked by OptiTrack. This transformation is performed in two steps:

1. **Translation Adjustment:** Adjust the coordinates by subtracting the translation vector $\mathbf{T}$, representing the participant’s position in the global space:

$$\mathbf{C}_{local} = \mathbf{C}_{global} - \mathbf{T}.$$

2. **Rotation Adjustment:** Rotate the data using the inverse of the rotation matrix R to align the gaze direction with the global coordinate system:

$$\mathbf{C}_{local} = R^{-1}\mathbf{C}_{local}.$$

This transformation ensures that the gaze direction vectors and the head planes are defined within the same global coordinate system, enabling accurate calculation of their intersections.

Intersection Calculation and Detection Once the gaze direction vectors and head planes are aligned in the global coordinate system, we calculate the intersection of a participant’s gaze vector $\mathbf{v}$ with the head plane of the other participant. The parametric equation for the gaze line is:

$$\mathbf{r}(t) = \mathbf{M} + t\mathbf{v},$$

where t is a scalar that defines the distance along the gaze vector. The intersection occurs when $\mathbf{r}(t)$ satisfies the plane equation, indicating that the gaze is directed at the other participant. For the gaze to be considered a direct look, the intersection point must lie within a radius of 2 cm from the other participant’s face. The 2 cm threshold was chosen based on the average size of human eyeballs, ensuring that the intersection point is close enough to be considered a direct gaze at the other participant’s face. This threshold ensures that gaze vectors intersect within realistic proximity, accounting for the spatial area around the eyes where mutual gaze typically occurs.

Simultaneous Gaze and Mutual Gaze Detection For an event to be classified as mutual gaze, both participants must be looking at each other simultaneously. This means that at the same time, both participants’ gaze vectors must intersect with the head plane of the other [70, 71]. To ensure temporal consistency, we apply a time-based smoothing technique that filters out brief, incidental gazes lasting less than 100 milliseconds. This ensures that only sustained eye contact events, indicative of meaningful social interactions, are classified as mutual gaze.

Handling Edge Cases In cases where the gaze vector does not intersect the other participant’s head plane or the intersection falls outside the 2 cm threshold, the event is not labeled as mutual gaze. If only one of them is looking at other participant’s eyes, we call this even direct gaze, and if direct gaze happens at the same time, it is a mutual gaze.

3.6 Data Collection Statistics

To better understand gaze behavior during interactive communication, we conducted two carefully designed experiments. Experiment 1 focused on how a physical barrier, such as a window, affects gaze and eye contact during face-to-face communication, while Experiment 2 extended this study by introducing a virtual communication condition through online meeting platform.

The key statistics for both experiments, including the number of participants, total recordings, and recording durations, are summarized in Table 3.2. After each session, all recordings were carefully reviewed to ensure the highest quality data. We selected the sessions with optimal tracking accuracy (for both gaze and head data), minimal data loss, and proper synchronization across systems.

Table 3.2: Summary of statistics from the two experiments, including participant count, session details, recording types, and total recording duration.

Statistic	Description
Total Participants	24 (Experiment 1), 24 (Experiment 2), 48 Total
Total Sessions	12 (Experiment 1), 12 (Experiment 2), 24 Total
Recording Types	Experiment 1: Face-to-Face, Over a Window; Experiment 2: Face-to-Face, Over a Window, online meeting platform
Recordings per Session	Experiment 1: 4 (2 Face-to-Face, 2 Over a Window); Experiment 2: 3 (1 Face-to-Face, 1 Over a Window, 1 online meeting platform)
Duration per Recording	8 minutes
Total Recordings	48 (Experiment 1), 36 (Experiment 2), 370 Total
Total Recording Hours	6.4 hours (Experiment 1), 4.8 hours (Experiment 2), 11.2 hours Total

3.7 Summary

This chapter outlined the process of collecting 3D gaze and head pose data for studying eye contact interactions between two participants.

Using OptiTrack motion capture system for head tracking and Tobii Pro Glasses 3 for gaze capture, we synchronized data and integrated it into a global coordinate system. Data collection included three conditions: face-to-face, through a window, and over online meeting platform. Mutual gaze detection involved identifying intersections between gaze vectors and head planes in the shared coordinate space. This dataset enables detailed analysis of eye contact interactions.

Chapter 4

Data Analysis

This chapter describes the data analysis process for examining eye contact behavior across various communication settings. The primary objective is to compare how eye contact differs between face-to-face interactions and virtual settings, such as Zoom, and to assess the impact of participant roles (questioner, responder) on these interactions.

4.1 Methodology of Data Analysis

This section describes the methodology employed to analyze eye contact behavior across various communication conditions in our study. The analysis in this chapter is based exclusively on data from Experiment 2, which includes recordings for all three communication scenarios: face-to-face, through a window, and via an online platform. By focusing on Experiment 2, as outlined in Section 3.6, we ensure a comprehensive and balanced comparison across these conditions.

4.1.1 Data Integration

To ensure consistency across all interaction setups, all data, including eye gaze and head orientation, were aligned and synchronized within a unified 3D coordinate system. As explained in the data collection chapter for the exact steps and procedure, each participant's face was represented as a plane within this system, serving as the basis for defining gaze events 3.2.

4.1.2 Identification of Eye Contact Events

Eye contact was defined differently based on the communication condition:

- **Face-to-Face and Screen-Sized Window Conditions:** In these settings, a participant's gaze vector was projected from the midpoint of their eyes toward the defined gaze point. Eye contact was identified when this vector intersected a 5 cm radius around the other participant's eye origin in the 3D space, indicating direct gaze toward the eyes (see 3.5 for a detailed explanation).
- **Zoom Condition:** Due to the screen-based nature of the Zoom condition, true eye contact does not occur as it does in face-to-face interactions. Instead, we defined "simultaneous direct gaze" to capture equivalent eye contact events. This was determined when participant A looked at participant B's eyes on the iPad screen while participant B simultaneously looked at participant A's eyes on their respective screen. This definition allowed us to meaningfully compare these gaze events with those observed in face-to-face conditions. For details about the calibration process of the Zoom camera and screen mapping, refer to Appendix A.3.

4.2 Data Analysis Results

Statistical analyses were conducted to identify significant differences in gaze behavior across the communication settings and participant roles. Comparisons were made to assess how the physical and virtual environments affected eye contact frequency and duration, highlighting the impact of screen-mediated communication on natural gaze dynamics. By frequency, we mean the proportion of time participants maintain eye contact during the interaction.

4.2.1 Effect of Condition

We analyzed the duration and frequency of direct gaze events, eye contact events, comparing these metrics across the three conditions. In this context, frequency represents the proportion of time participants spent looking at each other during the interaction.

As shown in Figure 4.1a, the direct gaze duration is highest in the window condition. Similarly, Figure 4.1b demonstrates that direct gaze frequency is comparable for face-to-face and window conditions but significantly lower for Zoom. In other words, this means that the size of the screen does not affect the duration and frequency of the direct gaze. However,

an online meeting changes this behavior and shows that the number of direct gaze events and the duration of these direct gazes are reduced while they are in an online meeting. A repeated measures ANOVA revealed a significant effect of condition on direct gaze duration, $F(2, 46) = 13.59, p < .001, \eta^2 = .238$, with significant differences between Zoom and the other conditions. Similarly, for direct gaze frequency, ANOVA results showed a significant effect of condition on direct gaze frequency, $F(2, 46) = 19.60, p < .001, \eta^2 = .228$, with Zoom having significantly lower frequencies than the other conditions.

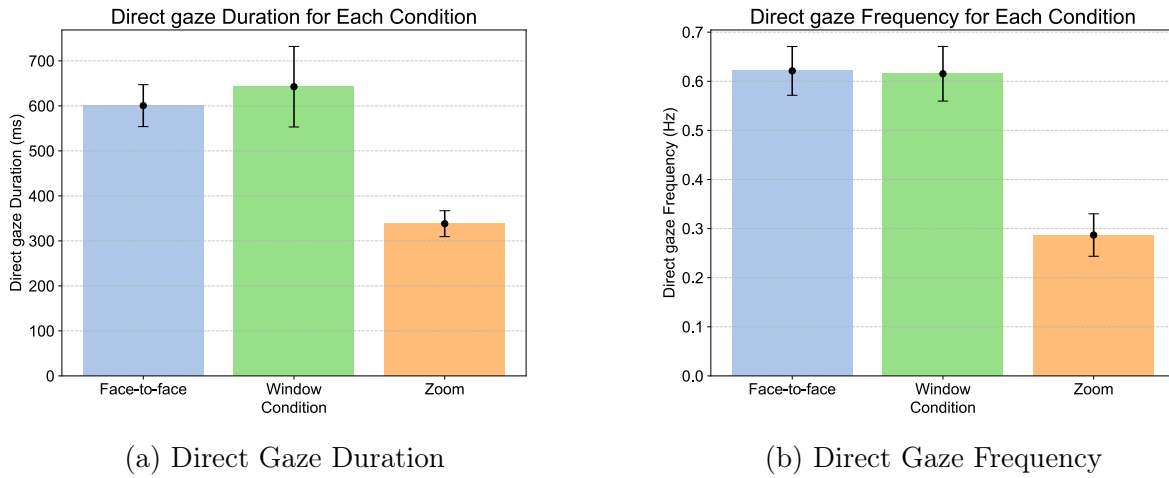
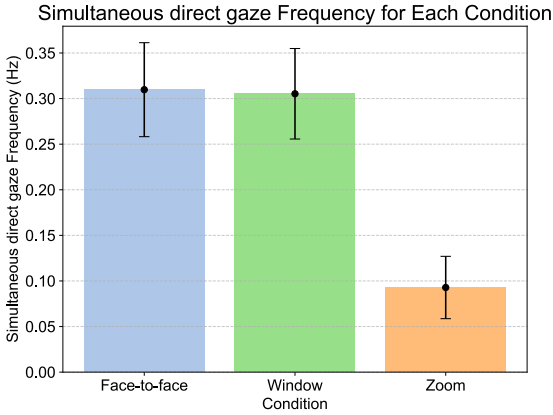
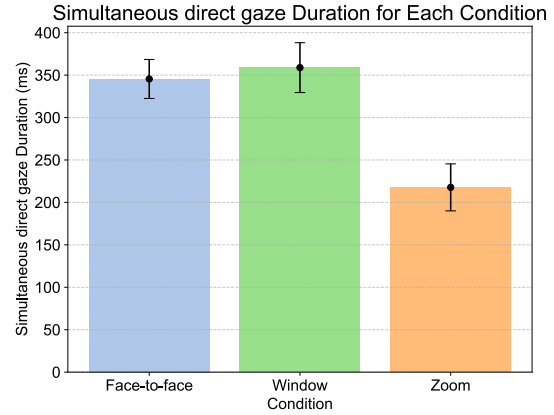


Figure 4.1: Direct gaze duration and direct gaze frequency across three conditions: face-to-face, screen size window, and Zoom.

The results in Figure 4.2 indicate that while face-to-face and window conditions exhibit a comparable simultaneous direct gaze frequency and duration, the Zoom condition shows a significant reduction, as detailed in Figures 4.2a and 4.2b, which again supports the claim that the online platform significantly affects the behavior of eye contact. Repeated measures ANOVA showed a significant effect of the condition on both the duration of simultaneous direct gaze, $F(2, 22) = 10.35, p < .001, \eta^2 = .32$, and the frequency, $F(2, 22) = 14.67, p < .001, \eta^2 = .290$. Zoom showed significantly lower values compared to face-to-face and window conditions.



(a) Simultaneous Direct Gaze Frequency



(b) Simultaneous Direct Gaze Duration

Figure 4.2: Simultaneous direct gaze frequency and simultaneous direct gaze duration across three conditions: face-to-face, screen size window, and Zoom. The results indicate that the screen size of Zoom does not significantly impact the measured values, suggesting similar interaction dynamics across screen-based settings, however there is a significant difference with zoom condition.

Figure 4.3 presents the probability distribution of direct gaze occurrences during dyadic interactions, comparing the likelihood of mutual eye contact to independent gaze actions. The observed probabilities in face-to-face settings indicating that gaze behavior is not random but coordinated and responsive to the interaction. Conversely, in the Zoom condition, the lower probabilities suggest that virtual setups disrupt this coordination, reducing the natural flow of mutual gaze. This highlights how environmental and technological factors shape gaze dynamics in social interactions.

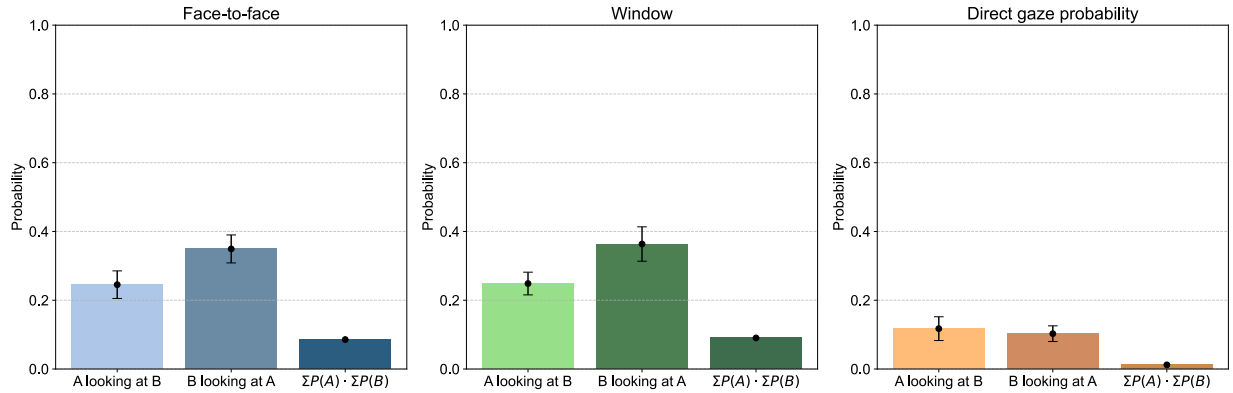


Figure 4.3: **Direct Gaze Probability.** This figure shows the probability distribution of direct gaze occurrences during dyadic interactions. The expression $\Sigma(p(A)) \times \Sigma(p(B))$ refers to the combined probability of both participants A and B engaging in gaze behaviors simultaneously, highlighting the likelihood of mutual eye contact compared to independent gaze actions.

The comparison in Figure 4.4 reveals that simultaneous direct gaze events occur less frequently than expected under the assumption of independent gaze behaviors. In simpler terms, people are not randomly making eye contact, and certain interaction dynamics seem to decrease the frequency of simultaneous direct gaze events compared to what would be expected if they were acting independently. This finding highlights the need for further exploration of how screen-mediated interactions alter the natural flow of gaze behavior.

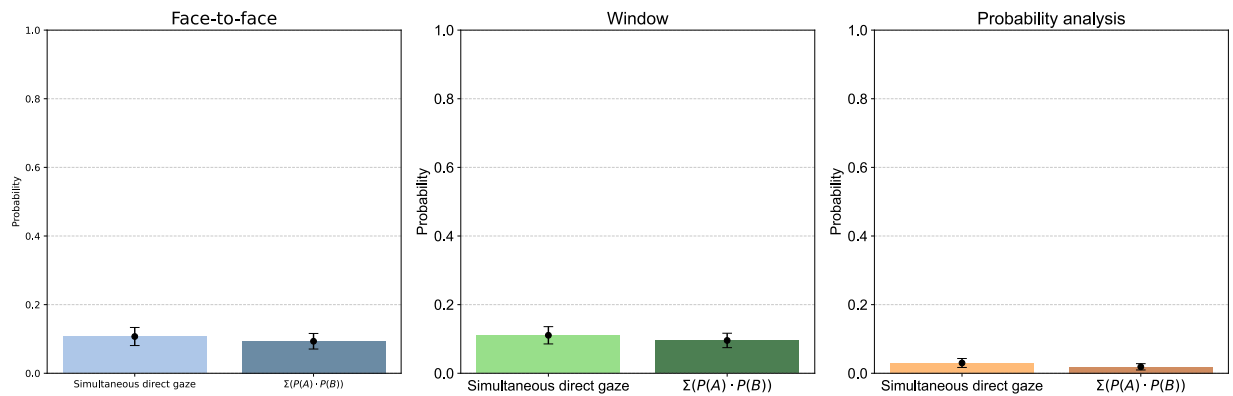


Figure 4.4: **Probability Analysis of Gaze Events.** This figure illustrates the comparison between the observed probability of simultaneous direct gaze events and the expected joint probability.

4.2.2 Effect of Role

The analysis in Figure 4.5 examines how the roles of questioner and responder influence gaze behavior in dyadic interactions. Here, frequency is defined as the proportion of time participants maintain direct gaze during the interaction. Responders maintain direct gaze more frequently, indicating attentiveness, while questioners often look away. These findings highlight the impact of role-based dynamics on eye contact, emphasizing the need to consider role when analyzing communication behaviors. Understanding these patterns can improve interaction models and insights into social behaviors in various settings.

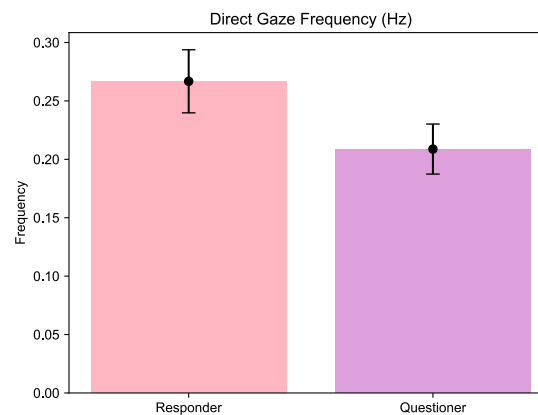


Figure 4.5: **Effect of Role on Gaze Behavior.** This figure shows the frequency of direct gaze behaviors categorized by the roles of participants during conversation. The data illustrates how the role (e.g., questioner vs. responder) influences gaze dynamics, highlighting differences in gaze frequency. The results indicate that questioners tend to have less direct gaze compared to responders.

4.3 Summary

This chapter analyzed eye contact behavior across three communication settings: face-to-face, face-to-face with a screen, and Zoom, using data collected during a Heads-Up game. By integrating eye gaze and head orientation into a unified 3D coordinate system, we defined eye contact events specific to each setup. For Zoom, we defined "simultaneous direct gaze" to be able to compare it with face to face eye contact events.

The findings highlighted significant differences in eye contact frequency and duration between in-person and virtual interactions. Role analysis showed that responders maintained

direct gaze more often than questioners. These insights reveal how both communication medium and participant roles influence eye contact behavior in social interactions.

Chapter 5

Modeling

5.1 Dyadic Model Properties

Generative models are important in machine learning because they learn patterns from data to create new, examples. Models like Generative Adversarial Networks (GANs) and Variational Autoencoders (VAEs) are used in many areas, such as creating images and videos, where understanding complex data patterns is crucial.

Among these, diffusion models have recently gained attention due to their impressive ability to generate high-quality, detailed outputs. Unlike other generative models that directly learn the data distribution, diffusion models approach the problem by progressively adding noise to the data through a forward process, transforming it into random noise. The reverse process then learns to denoise this noisy input step-by-step, reconstructing data points.

In addition to their high sample quality, we chose diffusion models for gaze modeling due to their proven success in related fields. For example, MDM (Human Motion Diffusion Model) [72] has shown how diffusion models can generate natural and expressive human motion, while MotionDiffuser [34] has demonstrated their capability in modeling multi-agent trajectories with permutation-invariant properties. These examples highlight the versatility of diffusion models in handling complex temporal and spatial data, making them ideal for our project’s goal of generating continuous, contextually accurate gaze behavior sequences that reflect real-world interactions.

Designing an effective diffusion architecture for modeling gaze behavior and head pose in a dyadic interaction setting requires addressing several unique challenges. These challenges guide the design choices for the architecture, ensuring it captures the real-world interactions between two agents.

- **Time Series with Two Agents:** The data we are modeling consists of time series that capture interactions between two agents, Person A and Person B, over a continuous time span. These interactions are characterized by strong temporal dependencies, where each time step is influenced by its neighbors, much like frames in a video. This correlation ensures continuity and reflects the natural flow of interactions.
- **Permutation Invariance:** A core requirement for our model is to ensure that it treats both agents symmetrically [73]. Permutation invariance in our model means that the probability distribution of the model’s predicted distribution remains the same even if the inputs for Person A and Person B are swapped. This property is crucial for preventing any bias in the model’s behavior that could arise from the ordering of the input data. Achieving permutation invariance requires architectural choices that guarantee symmetry in processing data from both agents, such as using shared network branches or specific attention mechanisms.
- **Permutation Invariance and Permutation Equivariance:** It is important to differentiate between permutation invariance and permutation equivariance. Permutation invariance that we desire for our model is the consistent probability output by it regardless of the order of the inputs, ensuring the model treats all agents symmetrically without any bias from input ordering. On the other hand, permutation equivariance refers to a property where a change in the input order (e.g., swapping Person A and Person B) results in a corresponding change in the output [74]. In this work, permutation equivariance in the model’s output is used to ensure that the distribution it represents is permutation invariant. Specifically, the diffusion model used produces scores that must be permutation equivariant to the order of the actors. Since the scores serve as updates to each agent’s features (such as position or gaze), the order of these scores must match the input order to ensure that the updates are applied to their corresponding actors.

These challenges guide the design of our architecture, leading us to include elements that ensure permutation invariance and equivariance while handling time-series data. This architecture can learn agent motion and gaze while inherently preserving the permutation invariant structure of the data.

5.2 Data Representation and Notation

In this project, we model head pose and gaze direction data to capture the interactions between two participants in a dyadic conversation. Our data is structured as time-series sequences that represent the physical orientation and movement of each participant’s head and eyes.

The full state of the data for a time window of size F is represented as a tuple $x = (P_1, P_2)$, where P_1 and P_2 correspond to the states of Person A and Person B, respectively. Each participant’s state over the time frames is stacked into a tensor $P_i \in \mathbb{R}^{F \times C}$, where F is the number of frames in the sequence, and C is the total number of channels representing the data at each frame. Overall, C is composed of the actor’s head rotation, position, and gaze directions for both eyes, making up a total of $4 + 3 + 3 + 3 = 13$ dimensions of C . For an eight-second sequence, F is set to 960 frames, given that the data is captured at 120 Hz.

To explain further, Head rotation requires four parameters because it is represented using quaternions, a mathematical representation that encodes 3D rotations. A quaternion q consists of four components: $q = (q_w, q_x, q_y, q_z)$, where q_w is the scalar (real) component representing the rotation angle, and q_x, q_y, q_z are the vector (imaginary) components representing the axis of rotation. Moreover, the head position requires three parameters because it represents a position in 3D space, described by the coordinates (x, y, z) .

For gaze direction, three parameters are sufficient to describe the line of sight in 3D space. This vector specifies the direction in which the participant is looking and is expressed using three coordinates (x, y, z) . These three components describe the orientation of the gaze relative to the participant’s head, allowing precise tracking of the participant’s focus in 3D environments.

In this study, we utilized a dataset comprising 56 recordings of face-to-face interactions, each lasting eight minutes, we exclude conversation over Zoom. Of these, 50 recordings (approximately 7 hours, accounting for 90% of the total data) were used for training, while 6 recordings (approximately 1 hour, representing 10% of the total data) were reserved for validation.

5.3 Modeling Eye-Contact

In this project, we used diffusion models to generate eye-contact behavior in a dyadic interaction setting. This required a non-standard model architecture to treat the actors

in a permutation invariant manner. We also model quantities that have representational constraints that ensure that they are meaningful. For example, we represent rotations using quaternions that have a unit-norm.

Head rotation quaternions are preferred over a direct 3D representation for training a diffusion model because they provide a robust and compact representation of rotations in 3D space while avoiding several key issues [75]. Unlike Euler angles, quaternions are not susceptible to gimbal lock, a phenomenon where certain rotations lead to a loss of one degree of freedom. Additionally, quaternions ensure smooth interpolation and are computationally efficient, which is crucial for a diffusion model that generates continuous transformations.

Compared to rotation matrices, quaternions are more compact, requiring only four parameters instead of nine, and inherently enforce the unit-norm constraint, simplifying regularization during training [76]. This unit-norm property ensures that quaternions always represent valid rotations, whereas 3D vectors or matrices may require additional normalization to maintain validity. As a result, quaternions are well-suited for modeling rotational dynamics in high-dimensional generative tasks such as those handled by diffusion models.

5.3.1 Adapting Diffusion Models for Eye-Contact Prediction

Conventional diffusion models typically do not account for permutation invariance in their input data [77]. For our task, where the model must symmetrically process two agents without bias, ensuring permutation invariance in the distribution is essential. This is achieved by incorporating architectural components that are permutation equivariant. Our model architecture draws inspiration from video diffusion models that use a U-Net framework with alternating space and time processing layers [78]. Figure 5.1 illustrates the detailed architecture of our model, including the multi-branch transformer design that processes gaze data symmetrically for both participants. In these layers serve two purposes:

- **Space layers** process each frame of a video independently, extracting spatial features from each frame without considering temporal dependencies.
- **Time layers** facilitate information exchange across frames, allowing the model to capture temporal dependencies and maintain coherence across time.

Adaptation for Dyadic Interaction Modeling In our approach, we used convolutional layers to independently process features from each actor. To exchange information between

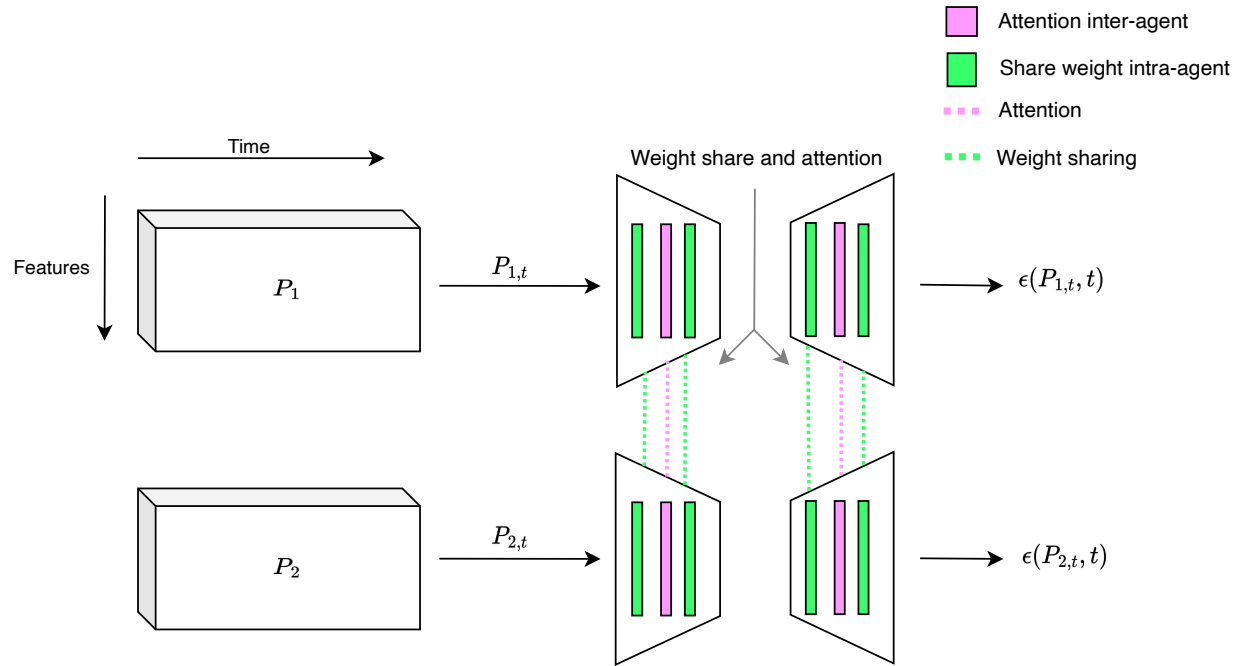


Figure 5.1: To ensure that the training process is permutation invariant, a detailed framework is shown here. We will use a multi-branch transformer, where each person’s gaze data will be processed through a separate branch.

the two agents in a permutation equivariant manner, we incorporated transformer layers [79]. These layers allow the model to symmetrically integrate data from both participants, preserving permutation invariance of the distribution by processing the features in a permutation equivariant manner.

5.3.2 Handling Constraints in Data Representation

Some components of our data, such as head rotation quaternions and gaze direction vectors, have inherent constraints (e.g., they are unit-norm). Although we did not impose hard constraints in the model, the generated outputs were observed to be approximately close to unit-norm.

Post-Processing Norm Adjustment To ensure that the generated quaternions and gaze vectors maintain unit norm, we applied an L_2 normalization after generation. This practice aligns with common approaches seen in prior works where generated data is post-processed to satisfy specific constraints [75].

5.3.3 Training Procedure

To prepare the model for training, input data such as positions were normalized to follow a standard normal distribution. This normalization step is required, as the diffusion model formulation is most optimal when the data has a unit normal distribution. During the normalization process, the mean and standard deviation of each feature were computed and saved. These parameters are retained and applied during the denormalization step after generation, allowing the model’s outputs to be accurately transformed back to their original scale and distribution. We specifically apply this normalization to the head positions since the position values are theoretically unbound.

Loss Function and Optimization The diffusion model was trained using a mean squared error (MSE) loss to guide the reconstruction of data during the denoising process, see section 2.4.1 for more details. The optimization was performed using the Adam optimizer [47].

Training Hyperparameters The training hyperparameters included a learning rate of 1×10^{-4} , a batch size of 64, and a total of 3000 epochs. The model architecture was configured with 6 transformer layers.

5.4 Summary

In this chapter, we developed a generative diffusion model for predicting eye-contact behavior in dyadic interactions. Key challenges included modeling time-series data with permutation invariance to handle two agents symmetrically. Our architecture combined convolutional layers for individual feature extraction and transformer layers for permutation equivariant information exchange. We addressed data constraints by applying L_2 normalization to maintain unit-norm properties for quaternions and gaze vectors. Training involved data normalization, MSE loss, and optimization with the Adam optimizer. Overall, the model generates gaze behavior while maintaining permutation invariance to symmetrically handle data from two agents.

Chapter 6

Results

This chapter presents the results of our generated gaze direction and head movements for two participants, evaluated both quantitatively and qualitatively. For the quantitative analysis, we assess the generated data using four levels of metrics: single-eye metrics using eye movement types like saccade and fixation characteristics, interocular metrics, head-eye correlation, and dyadic metrics such as eye contact duration. These metrics allow for a comprehensive comparison between the generated and real data, validating the synthetic outputs. For the qualitative analysis, the generated data were visualized using an avatar-based communication tool developed in Unity, and we visually evaluated the dynamics of the synthetic gaze and head movements.

6.1 Eye Movement Event Detection

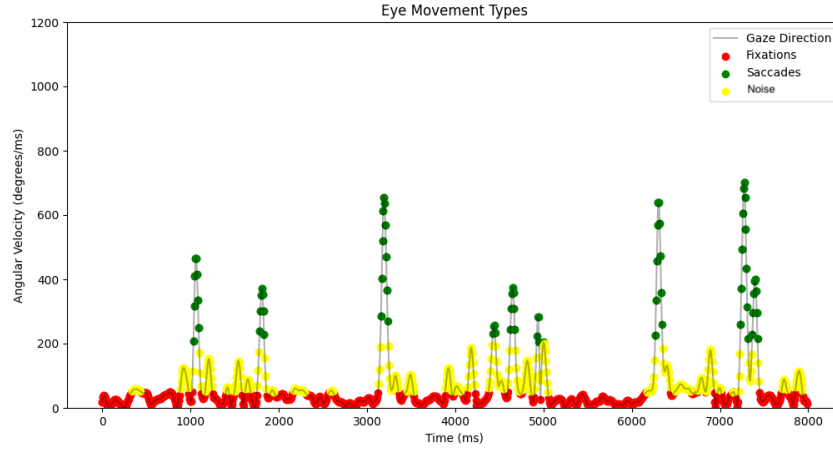
To evaluate our generated eye movement data, we needed to label different types of eye movements specifically fixations, saccades to analyze their frequency duration, and coordination with head movements. Although we initially considered established clustering-based methods from the literature for labeling eye movement types [80, 81, 82], we found that these algorithms were not sufficiently accurate for our data. Therefore, we employed alternative parameters based solely on dispersion metrics, such as velocity and acceleration.

The detection and labeling of eye movements were performed using a combination of velocity-based thresholds and angular displacement calculations [83]. This process involved analyzing the angular velocity and acceleration of gaze directions, as detailed below:

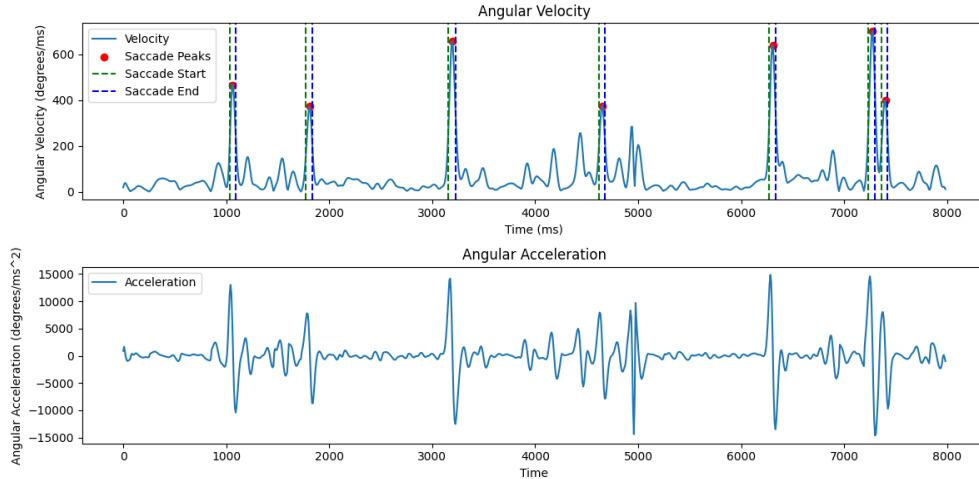
- **Data Filtering and Normalization:** Gaze direction data were smoothed using a Gaussian filter with a standard deviation (σ) of 1 to reduce noise and ensure consistent detection of eye movement types across frames.
- **Saccade Detection:** Saccades were identified using angular velocity data of the gaze direction. Peaks in angular velocity were detected with a threshold criterion set at 300 degrees per second, following the guidelines proposed by [83]. The detection process employed the *find_peaks* function from the SciPy library, which locates significant peaks based on the following criteria:
 - Peak Threshold: A peak was considered a potential saccade if the angular velocity exceeded 300 degrees per second.
 - Prominence Check: The prominence of each detected peak was calculated using *peak_prominences* to ensure that the peak stood out from its surrounding values. Only peaks with substantial prominence were classified as saccades.
 - Duration and Width: The width of the detected peak at half prominence was measured using the *peak_widths* function with *rel_height* = 0.5. This provided the start and end points for each saccade.
 - Classification: If any velocity within the detected peak’s start-to-end interval exceeded the 300 degree threshold, the interval was marked as a saccade in the *movement_types* array.
- **Fixation Identification:** Fixations were detected by locating intervals where the gaze direction remained stable. The process began by identifying segments labeled as potential fixations in the movement types array. Each potential fixation was checked for consistency by measuring the angular displacement between the start and end points of the interval. If this angular change exceeded a threshold of 1 degree, the segment was reclassified as non-fixation to ensure accuracy [83]. This approach helped in isolating true fixations from other movements by ensuring gazes experienced minimal movement throughout the fixation. The start and end of each fixation were also verified to ensure they adhered to these criteria. We also made sure that fixation are longer than a duration length of 100ms [83].

Figure 6.1 illustrates a sample of our labeling method. Figure 6.1(a) shows the angular velocities and the detected saccades, while Figure 6.1(b) displays the labeled sequence of eye movements.

These labeled movements were then used to calculate the frequency and duration of saccades and fixations for each eye separately, as well as to analyze the coordination between the eyes and with head movements.



(a) Labeled Eye Movement Types: The angular velocity of gaze direction over time is shown, with saccades, fixations, and noise regions identified. Peaks in angular velocity correspond to saccades, while stable intervals indicate fixations.



(b) Angular Velocity and Acceleration: The detailed angular velocity and acceleration data of eye movements are plotted, highlighting peaks and thresholds used for movement classification.

Figure 6.1: A visualization of the labeling process for eye movements, showing how angular velocity and acceleration metrics were utilized to classify fixations, saccades, and noise.

6.2 Quantitative Results

The quantitative analysis focused on several levels of gaze and head movement coordination. The fixation and saccade metrics reported in studies involving tasks such as observing static images or art pieces [84, 85] often yield different durations and frequencies compared to those seen in conversational interactions. In these controlled settings, participants’ eye movements are influenced by the nature of the task, which usually requires longer periods of fixation. In contrast, natural conversations involve more frequent and rapid shifts in gaze due to social cues and interaction dynamics.

To place our findings in a more appropriate context, we refer to literature that measures eye movement metrics during active tasks [83, 86], such as walking or performing physical activities. Metrics from these studies indicate that eye movement during active tasks features shorter and more varied fixations, which aligns more closely with the patterns observed in our generated data. This approach underscores the novelty of our task and emphasizes the challenge in directly benchmarking against existing literature.

Z-Score Definition and Interpretation To compare the similarity between the generated and real data for metrics such as saccade frequency, fixation duration, and others, we compute the Z-score of the differences in their mean values. The Z-score is defined as:

$$Z = \frac{|\mu_1 - \mu_2|}{\sqrt{\frac{\sigma_1^2 + \sigma_2^2}{2}}},$$

where μ_1 and μ_2 represent the mean values of the real and generated data, respectively, and σ_1^2 and σ_2^2 are their corresponding variances. The Z-score quantifies the normalized distance between the means of two datasets relative to the pooled standard deviation. A smaller Z-score indicates greater similarity between the two distributions, suggesting that the generated data closely matches the real data.

6.2.1 Single Eye Analysis

Saccade frequency and duration are critical metrics as they reflect the dynamism and responsiveness of eye movements, indicating the model’s ability to simulate natural gaze shifts. Table 6.1 presents a comparison of these metrics between the real and generated data, showing the mean values and standard deviations. Similarly, fixation metrics are essential

as they serve as the simple statistics to identify plausible eye movements. Metrics such as saccade frequency and fixation duration were computed for each eye separately.

Table 6.1: Comparison of mean values for fixation frequency, fixation duration, saccade frequency, and saccade duration between generated and real data, with corresponding standard deviations shown as $\pm$ values. Lower Z-scores indicate greater similarity between the generated and real data, reflecting the model’s effectiveness in replicating realistic eye movement patterns.

Metric	Real Data	Generated Data	Z-Score
Saccade Duration (ms)	41 ± 10	35 ± 15	0.42
Saccade Frequency	0.2 ± 0.153	0.3 ± 0.15	0.66
Fixation Duration (ms)	350 ± 100	400 ± 150	0.39
Fixation Frequency	0.6 ± 0.184	0.5 ± 0.214	0.48

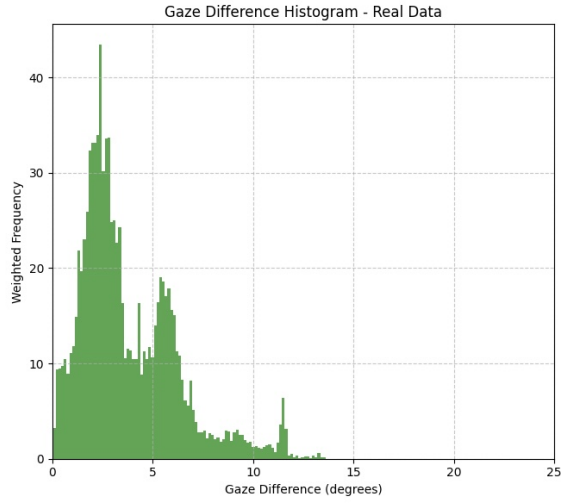
6.2.2 Interocular Eye Coordination (Vergence)

The vergence between the left and right eyes was analyzed by calculating the angular difference between their respective gaze directions. Vergence, denoted as θ , quantifies the coordination of the two eyes as they converge or diverge to focus on a point in 3D space. It was calculated using the following formula:

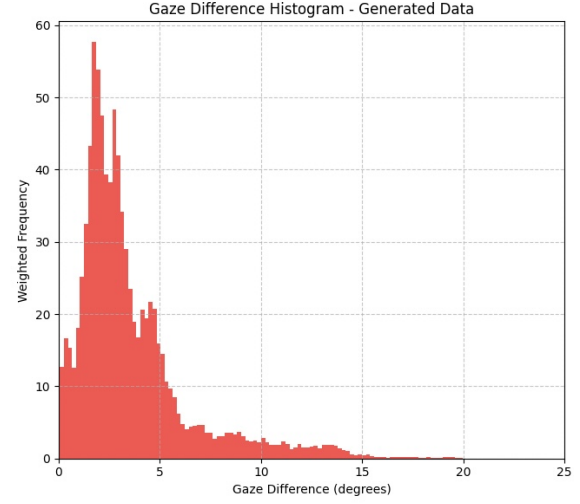
$$\theta = \cos^{-1} \left(\frac{\mathbf{v}_1 \cdot \mathbf{v}_2}{\|\mathbf{v}_1\| \|\mathbf{v}_2\|} \right),$$

where $\mathbf{v}_1$ and $\mathbf{v}_2$ are the gaze direction vectors for the left and right eyes, respectively. The normalized dot product $\frac{\mathbf{v}_1 \cdot \mathbf{v}_2}{\|\mathbf{v}_1\| \|\mathbf{v}_2\|}$ computes the cosine difference between the directions, which is then converted to degrees using $\cos^{-1}(\cdot)$.

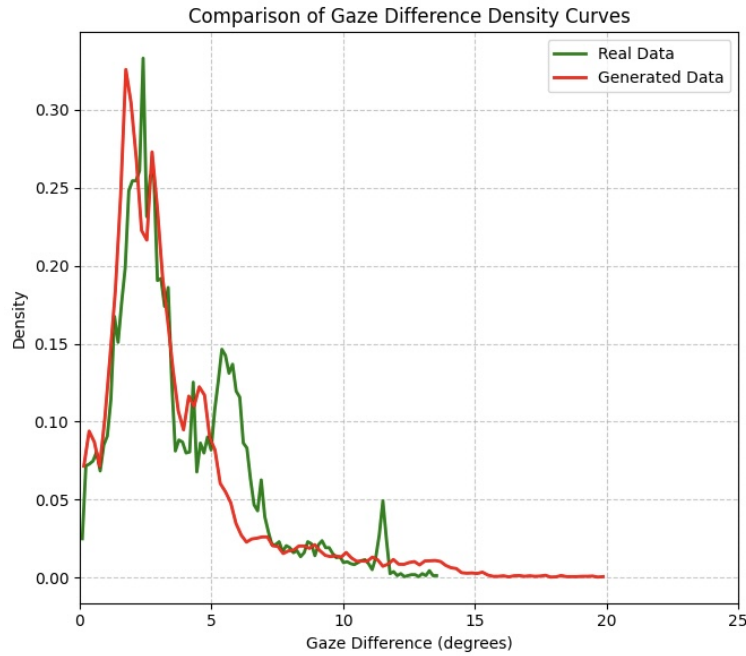
This analysis involved computing the distribution of angular differences across all gaze samples. The results are visualized in three plots in figure 6.2: (a) the distribution of angular differences for all real data, (b) the distribution of angular differences for the generated data, (c) the outlines of the density curves for both real and generated data distributions, overlayed for ease of visual comparison. These density plots represent the probability distribution of angular differences, indicating how frequently specific angular differences occur within each dataset. The similarity seen in plot (c) shows the model’s good performance generating interocular eye movements.



(a) Average Gaze Difference Distribution for Real Data



(b) Average Gaze Difference Distribution for Generated Data



(c) Comparison of Density Curves for Real and Generated Data

Figure 6.2: Comparison of real and generated gaze difference distributions. The third plot overlays the outlines of the first two plots to facilitate a qualitative comparison between the distributions. The frequency represents the number of frames with a specific angular difference in degrees.

To quantitatively evaluate the interocular eye coordination between the real and generated data, we computed Earth Mover’s Distance (EMD) [87] between their gaze difference distributions. In figure 6.2 we show the result. EMD is a measure used to quantify the difference between two probability distributions. In essence, it can be thought of as the minimal amount of “work” required to transform one distribution into the other, where “work” is defined as moving a certain amount of probability mass a given distance. Smaller EMD values indicate that the two distributions are more similar.

In this analysis, the EMD value of 0.54 degrees indicates that, on average, only a small displacement is required to align the generated gaze difference distribution with the real gaze difference distribution. This low value demonstrates a close match between the two distributions, confirming that the generated eye movements accurately replicate realistic interocular coordination. This finding highlights the effectiveness of our model in capturing and mimicking human visual behavior.

6.2.3 Eye-Head Coordination

Eye-head coordination was evaluated to examine how gaze and head movements were synchronized to maintain stable vision during head turns.

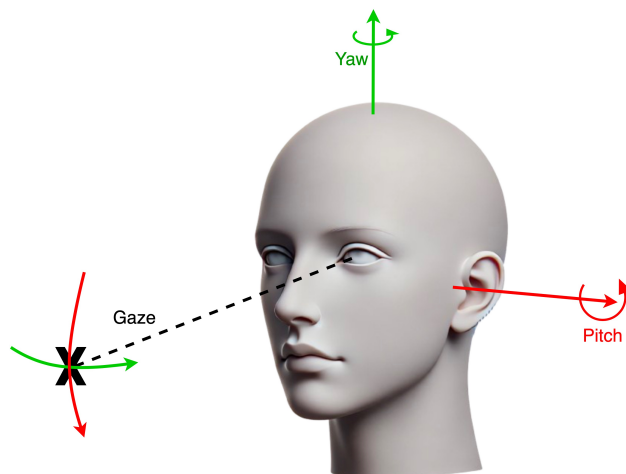


Figure 6.3: The yaw represents horizontal rotation of the head, the pitch indicates vertical rotation, and the gaze direction (black dashed line) shows the line of sight extending from the eyes and its rotations.

The coordination between head and eye movements was analyzed by calculating the correlation between their angular positions (yaw and pitch). As shown in Figure 6.3, the yaw represents horizontal rotations (e.g., turning the head left or right), while the pitch represents vertical rotations (e.g., looking up or down). Roll was excluded from our analysis because it primarily involves tilting the eye or head to the side, stabilizing the visual field during head tilts but is less relevant for understanding how the eyes and head work together to direct gaze.

To compute these correlations, the head’s rotational data was extracted and converted to yaw and pitch angles using quaternion transformations, while the gaze direction vectors were similarly processed to derive corresponding yaw and pitch angles. Saccadic moments were excluded from the analysis as they involve rapid eye movements that are not related to head coordination [88] and could distort the results.

Table 6.2: Correlation values comparing the relationships between head yaw and gaze yaw, as well as head pitch and gaze pitch, in both real and generated datasets.

Feature 1	Feature 2	Real Data Correlation	Generated Data Correlation
Head Yaw	Gaze Yaw	-0.09 ± 0.02	-0.04 ± 0.01
Head Pitch	Gaze Pitch	-0.08 ± 0.015	-0.06 ± 0.01

The Pearson correlation coefficient was used to quantify the relationship between the angular trajectories of head and gaze. The correlations for both real and generated data, shown in Table 6.2, are close to zero.

This result can be explained by the opposing contributions of different coordination behaviors. Positive correlations occur when the head and eyes rotate in the same direction, such as when noticing something or responding to a call, where both the head and gaze align to focus on the target. Conversely, negative correlations arise during the Vestibulo-Ocular Reflex (VOR) [89], where eye movements compensate for head movements to stabilize vision during head turns. These opposing effects effectively neutralize each other, resulting in an overall correlation close to zero.

This finding aligns with natural eye-head coordination, where behaviors like the VOR and simultaneous head-gaze movements interact dynamically, balancing each other in the observed data.

6.2.4 Dyadic Coordination

At the interaction level, the analysis focused on key metrics such as eye-contact frequency, reflecting the mutual engagement between participants. These metrics were crucial for validating the social realism of the generated interactions, highlighting the role of gaze in communication.

Table 6.3: Comparison of eye contact metrics between real and generated data, including eye contact frequency, frequencies of A looking at B and B looking at A, and joint probability of mutual gaze, with corresponding standard deviations shown as $\pm$ values. Z-scores indicate the normalized distance between the means of the real and generated data.

Metric	Real Data	Generated Data	Z-Score
Eye Contact Frequency	0.25 ± 0.106	0.6 ± 0.233	2.2
A Looking at B Frequency	0.45 ± 0.15	0.7 ± 0.264	1.41
B Looking at A Frequency	0.52 ± 0.195	0.65 ± 0.173	0.65
Joint Probability	0.23 ± 0.09	0.58 ± 0.154	0.25

The generated data have higher frequencies of eye contact compared to real interactions, as shown in table 6.3, in which the high Z-score also reflects a significant difference between the real and generated data. The main reasons for this discrepancy could be:

- **Lack of Social Context:** The model does not incorporate contextual factors like speech, emotions, or interaction dynamics that significantly impact eye contact frequency and patterns. Without this information, the model may default to generating more frequent eye contact than naturally occurs.
- **Training Data Constraints:** If the training dataset lacks diversity in eye contact behaviors or overrepresents certain patterns, the model may learn to produce those patterns excessively, leading to unrealistic frequencies in the generated data.

6.3 Qualitative Results

The qualitative analysis involved reviewing the generated data using our avatar-based application. Figure 6.4 shows the avatars used for this purpose, representing two participants in a conversation setting. This allowed us to visually see how closely the generated gaze and head

movements matched natural human behavior. The avatar movements appeared appropriate for a conversation setting, showing transitions between fixations, saccades, and head turns, and effectively representing the nature of interactions between two people positioned face-to-face. This visual review indicates that the generative model learned to simulate coordinated eye and head movements that align with social communication scenarios. The slight, synchronized eye movements observed align with our findings on interocular Eye Coordination, confirming that the model captured these interactions effectively. You can view sample videos through this link: <https://drive.google.com/drive/folders/13JNNtsH6KZLCUk6d4eE-RHAlMRJBHILV>.



(a) First Person



(b) Second person

Figure 6.4: Avatars that we used to visualize generated data

To provide further insights into the qualitative results, we include specific examples from the generated data:

- In **sample_1.mov**, some interesting head movements are observed, such as looking down. This behavior was frequently seen in the real data as participants often looked at the cards they were holding during the Heads-Up game. Additionally, the sample includes good fixations and saccades that last for durations consistent with real data.

- In **sample_2.mov**, both avatars (A and B) exhibit saccades where their eyes are actively looking around or down. This behavior mirrors natural saccades in conversation settings.
- In **sample_3.mov**, the male avatar performs an upward saccade toward the other avatar at second 3, and at second 1, exhibits a lateral saccade to the sides.
- In **sample_4.mov**, the avatars demonstrate shorter saccades that are consistent with expectations based on real data.
- For comparison, segments of the real dataset have been uploaded (**real_data_1.mov** and **real_data_2.mov**) to validate that the generated data has learned key features from the real recordings.

6.4 Summary

The results indicate that the generative model produces gaze and head movement patterns that align with some of expected behaviors, supported by both qualitative observations and quantitative metrics. Our visual tool allowed for an examination of how the generated movements captured interaction dynamics, suggesting potential use in simulations of social communication. The analysis of eye and head coordination, along with relevant interaction metrics, highlights the model's ability to represent generated behavior on the avatars.

Chapter 7

Discussion

As outlined in the introduction, the primary objective of this work was to address the challenge of generating gaze and head movements for dyadic interactions. This research aimed to fill the gap in the existing literature by introducing a generative model capable of producing gaze behaviors supported by a novel, dyadic synchronized dataset of people in a conversational setting. The discussion presented here reflects on the achievements of this research, its limitations, and potential implications while proposing future directions to enhance the field further.

7.1 Achievements

7.1.1 Novel Dataset for Dyadic Gaze and Head Movement

One of our work’s significant contributions is the development of a dataset capturing the synchronized eye and head movements of two participants engaged in natural conversations across three distinct settings: face-to-face, through a screen-sized window, and over Zoom. Unlike existing datasets, which often lack ground truth for mutual gaze dynamics or focus on single participants, this dataset provides precise and reliable data for studying real-time social interactions. The inclusion of diverse scenarios enhances its relevance for applications ranging from virtual communication tools to social psychology research.

7.1.2 Realistic Generative Model

The diffusion-based generative model developed in this thesis demonstrated the capability to replicate key aspects of human-like gaze behavior. Quantitative analysis revealed that generated data closely aligned with real data across several metrics, including fixation and saccade frequencies, durations, and head-eye coordination. The integration of transformer-based architectures allowed the model to capture temporal dependencies effectively.

7.1.3 Validation through Metrics and Visualization

The model was evaluated using both quantitative metrics and qualitative assessments. Quantitative comparisons showed low Z-scores for most metrics, indicating high similarity between real and generated data. Qualitative validation using avatar-based visualization confirmed the similarity of simulated gaze to real data and head movements, providing further confidence in the model’s performance.

7.2 Limitations

7.2.1 Simplified Interaction Scenarios

Data collection was conducted using a structured game-based setup, which, while effective for ensuring consistency, may not capture the full range of behaviors observed in everyday conversations. Spontaneous, unscripted interactions often involve more complex gaze patterns influenced by emotional cues, speech, and cultural norms. As a result, the model may not fully generalize to naturalistic settings.

7.2.2 Lack of Contextual Awareness

The model operates without incorporating contextual information, such as speech signals, facial expressions, or conversational dynamics. These factors play a crucial role in determining gaze behavior, and their absence limits the model’s ability to produce context-sensitive interactions.

7.2.3 Overestimated Eye Contact Frequency

The generated data exhibited higher frequencies of eye contact than real-world interactions, as evidenced by the Z-scores for eye contact metrics. This overestimation likely stems from the absence of social context in the model, resulting in more frequent and less nuanced gaze behaviors than natural interactions.

7.3 Future Work

To address these limitations and further enhance the model’s capabilities, the following directions are proposed for future research:

- **Expand the Dataset:**
 - Collect data from participants across diverse demographics, cultures, and social contexts to account for variations in gaze behavior.
 - Include unscripted conversations and varied interaction modalities like group discussions to better capture real-world complexities.
 - Enhance generalizability by reducing biases and enabling applications in cross-cultural studies and accessibility tools.
- **Incorporate Contextual Information:**
 - Add modalities like speech signals, facial expressions, and conversation scripts to provide context for gaze behavior.
 - Use multi-modal embeddings (e.g., CLIP) to refine interactions and make gaze dynamics responsive to emotional and conversational cues.
 - Improve model applications in empathetic AI, social robotics, and personalized communication systems.
- **Enhance Model Architecture for Real-Time Performance:** To enable integration into live VR/AR systems and telecommunication platforms, future work should focus on optimizing the computational efficiency of diffusion models. Key strategies include:
 - **Model Optimization:** Employ techniques like model pruning [90], weight quantization [91], and knowledge distillation [92] to reduce the model’s size and computational complexity while maintaining performance.

- Efficient Sampling: Investigate accelerated sampling methods for diffusion models, such as DDIM (Denoising Diffusion Implicit Models) [93], to reduce the number of steps required for generating outputs, thereby significantly improving latency.
- Model Parallelism: Develop architectures that distribute the computational load across multiple devices or processing units, reducing bottlenecks and achieving low-latency inference.
- Sparse Attention Mechanisms: Integrate sparse or adaptive attention techniques [94] into transformer components to handle long temporal sequences more efficiently.

7.4 Societal Impact

The generated gaze data have significant implications across various domains, offering transformative potential in technology, education, psychology, and accessibility. These models can revolutionize virtual and augmented reality environments in human-computer interaction by fostering more natural and engaging social interactions. By accurately replicating gaze behavior, these systems can enhance the sense of presence and realism in virtual meetings, online classrooms, and gaming, improving user experiences and productivity in remote communication scenarios. In education, the ability to simulate realistic social behaviors can serve as a valuable tool for training, such as teaching interpersonal skills, public speaking, or customer service techniques. Using gaze-driven avatars, learners can practice these skills in a controlled environment and receive immediate feedback.

In psychology, models provide researchers with a controlled platform to study gaze dynamics and their impact on communication, social bonding, and emotional expression, especially in virtual and online meetings. This opens new avenues for exploring human behavior in naturalistic and experimental settings.

However, while the potential benefits are immense, it is important to acknowledge the possible negative consequences. Misuse of such models in manipulative applications could raise ethical concerns. For example, replicating realistic gaze behavior might be used to have more realistic fake videos. Additionally, over-reliance on such technologies could inadvertently reduce human-to-human interactions in specific contexts, potentially affecting social development and emotional well-being. It is crucial to implement ethical guidelines and ensure transparency in developing and deploying gaze modeling technologies.

7.5 Conclusion

This study demonstrated the feasibility of using generative models to simulate realistic gaze and head movements for avatar-based communication tools. By collecting a synchronized dataset of eye and head movements during dyadic interactions, we established a foundation for modeling the dynamics of mutual gaze and eye contact. The diffusion model employed in this work generated synthetic movements that, despite certain limitations, captured key aspects of human interaction from collected data.

Our study underscores the importance of addressing the limitations in data collection, contextual modeling, and interaction diversity to achieve realism and generalizability. These findings pave the way for future research to enhance generative models, integrate multi-modal data, and expand datasets for broader applicability.

Ultimately, this work provides a stepping stone toward creating interactive systems that mimic natural social behaviors, improving user experience in virtual and augmented reality environments, and advancing our understanding of gaze-based interactions in human communication.

Bibliography

- [1] M. Argyle, M. Cook, and D. Cramer, “Gaze and mutual gaze,” *BJPsych*, no. 6, pp. 848–850, 1994.
- [2] C. L. Kleinke, “Gaze and eye contact: A research review.” *Psychol. Bull.*, no. 1, p. 78, 1986.
- [3] R. Canales, E. Jain, and S. Jörg, “Real-time conversational gaze synthesis for avatars,” in *MIG*, 2023, pp. 17:1–17:7.
- [4] Y. Song and S. Ermon, “Generative modeling by estimating gradients of the data distribution,” *CoRR*, 2019.
- [5] Tobii, “How Do Tobii Eye Trackers Work?” https://connect.tobii.com/s/article/How-do-Tobii-eye-trackers-work?language=en_US, January 2024, accessed: September 25, 2022.
- [6] NaturalPoint, Inc., “Motion capture provider- OptiTrack Motion Capture System,” <https://www.naturalpoint.com>, accessed: April 12, 2023.
- [7] E. Gilboa-Schechtman and I. Shachar-Lavie, “More than a face: A unified theoretical perspective on nonverbal social cue processing in social anxiety,” *Front. Hum. Neurosci.*, p. 904, 2013.
- [8] G. A. Holleman, R. S. Hessels, C. Kemner, and I. T. C. Hooge, “Implying social interaction and its influence on gaze behavior to the eyes,” *PLoS One*, no. 2, 2020.
- [9] D. DeSteno, C. Breazeal, R. H. Frank, D. Pizarro, J. Baumann, L. Dickens, and J. J. Lee, “Detecting the trustworthiness of novel partners in economic exchange,” *Psychol. Sci.*, no. 12, pp. 1549–1556, 2012.

- [10] J. J. Lee, C. Breazeal, and D. DeSteno, “Role of speaker cues in attention inference,” *Frontiers Robotics AI*, p. 47, 2017.
- [11] L. S. Bohannon, A. M. Herbert, J. B. Pelz, and E. M. Rantanen, “Eye contact and video-mediated communication: A review,” *Displays*, no. 2, pp. 177–185, 2013.
- [12] J. O. Hietanen and J. K. Hietanen, “Genuine eye contact elicits self-referential processing,” *Conscious. Cogn.*, pp. 100–115, 2017.
- [13] N. F. Troje, “Depth from motion parallax: Deictic consistency, eye contact, and a serious problem with zoom,” *J. Vis.*, no. 10, pp. 1–1, 2023.
- [14] G. Delgado, “Nvidia broadcast 1.4 adds eye contact and vignette effects with virtual background enhancements,” January 2023. [Online]. Available: <https://www.nvidia.com/en-us/geforce/news/jan-2023-nvidia-broadcast-update/>
- [15] NVIDIA, “Nvidia maxine-ar-sdk,” 2023, gitHub repository. [Online]. Available: <https://github.com/NVIDIA/MAXINE-AR-SDK>
- [16] A. Inc., “Arkit - integrate hardware sensing features to produce augmented reality apps and games,” 2024. [Online]. Available: <https://developer.apple.com/augmented-reality/arkit/>
- [17] Y. Mitsuzumi, A. Nakazawa, and T. Nishida, “DEEP eye contact detector: Robust eye contact bid detection using convolutional neural network,” in *BMVC*, 2017.
- [18] P. N. Belhumeur, D. W. Jacobs, D. J. Kriegman, and N. Kumar, “Localizing parts of faces using a consensus of exemplars,” *IEEE Trans. Pattern Anal. Mach. Intell.*, no. 12, pp. 2930–2940, 2013.
- [19] V. Le, J. Brandt, Z. Lin, L. D. Bourdev, and T. S. Huang, “Interactive facial feature localization,” in *ECCV*, 2012.
- [20] M. Köstinger, P. Wohlhart, P. M. Roth, and H. Bischof, “Annotated facial landmarks in the wild: A large-scale, real-world database for facial landmark localization,” in *ICCV Workshops*, 2011, pp. 2144–2151.
- [21] C. Sagonas, E. Antonakos, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic, “300 faces in-the-wild challenge: database and results,” *Image Vis. Comput.*, pp. 3–18, 2016.

- [22] D. Zhang, B. Wang, G. Wang, Q. Zhang, J. Zhang, J. Han, and Z. You, “Onfocus detection: identifying individual-camera eye contact from unconstrained images,” *Sci. China Inf. Sci.*, no. 6, pp. 1–12, 2022.
- [23] E. Chong, E. Clark-Whitney, A. Southerland, E. Stubbs, C. Miller, E. L. Ajodan, M. R. Silverman, C. Lord, A. Rozga, R. M. Jones *et al.*, “Detection of eye contact with deep neural networks is as accurate as human experts,” *Nat. Commun.*, no. 1, p. 6386, 2020.
- [24] T. Salzmann, B. Ivanovic, P. Chakravarty, and M. Pavone, “Trajectron++: Dynamically-feasible trajectory forecasting with heterogeneous data,” in *ECCV*, 2020.
- [25] S. Shahriar and Y. Kim, “Audio-visual emotion forecasting: Characterizing and predicting future emotion using deep learning,” in *FG*, 2019.
- [26] P. Müller, E. Sood, and A. Bulling, “Anticipating averted gaze in dyadic interactions,” in *ETRA*, 2020, pp. 7:1–7:10.
- [27] A. Graves, “Long short-term memory,” in *Supervised Sequence Labelling with Recurrent Neural Networks*, 2012.
- [28] P. Blache, M. Abderrahmane, S. Rauzy, and R. Bertrand, “An integrated model for predicting backchannel feedbacks,” in *IVA*, 2020, pp. 6:1–6:3.
- [29] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese, “Social LSTM: human trajectory prediction in crowded spaces,” in *CVPR*, 2016, pp. 961–971.
- [30] A. Gupta, J. Johnson, L. Fei-Fei, S. Savarese, and A. Alahi, “Social GAN: socially acceptable trajectories with generative adversarial networks,” in *CVPR*, 2018, pp. 2255–2264.
- [31] Y. Yuan, X. Weng, Y. Ou, and K. Kitani, “Agentformer: Agent-aware transformers for socio-temporal multi-agent forecasting,” in *ICCV*, 2021, pp. 9793–9803.
- [32] H. Huang, S. Kimura, K. Kuwabara, and T. Nishida, “Generation of head movements of a robot using multimodal features of peer participants in group discussion conversation,” *Multimodal Technol. Interact.*, no. 2, p. 15, 2020.
- [33] F. B. Tesema, J. Gu, W. Song, H. Wu, S. Zhu, Z. Lin, M. Huang, W. Wang, and R. Kumar, “Addressee detection using facial and audio features in mixed human–human

- and human-robot settings: A deep learning framework,” *IEEE Syst. Man Cybern. Mag.*, no. 2, pp. 25–38, 2023.
- [34] C. M. Jiang, A. Cornman, C. Park, B. Sapp, Y. Zhou, and D. Anguelov, “Motiondiffuser: Controllable multi-agent motion prediction using diffusion,” in *CVPR*, 2023, pp. 9644–9653.
- [35] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. C. Courville, and Y. Bengio, “Generative adversarial networks,” *Commun. ACM*, no. 11, pp. 139–144, 2020.
- [36] C. Saharia, W. Chan, H. Chang, C. A. Lee, J. Ho, T. Salimans, D. J. Fleet, and M. Norouzi, “Palette: Image-to-image diffusion models,” in *SIGGRAPH*, 2022, pp. 15:1–15:10.
- [37] J. Ho, T. Salimans, A. A. Gritsenko, W. Chan, M. Norouzi, and D. J. Fleet, “Video diffusion models,” in *NeurIPS*, 2022.
- [38] Y. Tashiro, J. Song, Y. Song, and S. Ermon, “CSDI: conditional score-based diffusion models for probabilistic time series imputation,” in *NeurIPS*, 2021, pp. 24 804–24 816.
- [39] A. van den Oord, N. Kalchbrenner, L. Espeholt, K. Kavukcuoglu, O. Vinyals, and A. Graves, “Conditional image generation with pixelcnn decoders,” in *NIPS*, 2016, pp. 4790–4798.
- [40] G. Papamakarios, E. T. Nalisnick, D. J. Rezende, S. Mohamed, and B. Lakshminarayanan, “Normalizing flows for probabilistic modeling and inference,” *CoRR*, 2019.
- [41] D. P. Kingma, “Auto-encoding variational bayes,” *arXiv preprint arXiv:1312.6114*, 2013.
- [42] L. Yang, Z. Zhang, Y. Song, S. Hong, R. Xu, Y. Zhao, Y. Shao, W. Zhang, M. Yang, and B. Cui, “Diffusion models: A comprehensive survey of methods and applications,” *CoRR*, 2022.
- [43] C. Meng, Y. He, Y. Song, J. Song, J. Wu, J. Zhu, and S. Ermon, “Sdedit: Guided image synthesis and editing with stochastic differential equations,” in *ICLR*, 2022.
- [44] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole, “Score-based generative modeling through stochastic differential equations,” in *ICLR*, 2021.

- [45] J. Ho, A. Jain, and P. Abbeel, “Denoising diffusion probabilistic models,” in *NeurIPS*, 2020.
- [46] L. Yang, Z. Zhang, Y. Song, S. Hong, R. Xu, Y. Zhao, W. Zhang, B. Cui, and M. Yang, “Diffusion models: A comprehensive survey of methods and applications,” *ACM Comput. Surv.*, no. 4, pp. 105:1–105:39, 2024.
- [47] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in *ICLR (Poster)*, 2015.
- [48] S. Mondal, Z. Yang, S. Ahn, D. Samaras, G. J. Zelinsky, and M. Hoai, “Gazeformer: Scalable, effective and fast prediction of goal-directed human attention,” in *CVPR*, 2023, pp. 1441–1450.
- [49] D. Tu, X. Min, H. Duan, G. Guo, G. Zhai, and W. Shen, “End-to-end human-gaze-target detection with transformers,” in *CVPR*, 2022, pp. 2192–2200.
- [50] M. K. X. J. Pan, S. Choi, J. Kennedy, K. McIntosh, D. C. Zamora, G. Niemeyer, J. Kim, A. Wieland, and D. L. Christensen, “Realistic and interactive robot gaze,” in *IROS*, 2020, pp. 11 072–11 078.
- [51] OptiTrack, “Primex 41 camera - specifications and pricing,” 2022. [Online]. Available: <https://optitrack.com/cameras/primex-41/>
- [52] —, “Calibration - optitrack motive documentation,” 2024. [Online]. Available: <https://docs.optitrack.com/motive/calibration>
- [53] —, “Rigid body tracking - optitrack motive documentation,” 2024, accessed: 2024-09-23. [Online]. Available: <https://docs.optitrack.com/motive/rigid-body-tracking>
- [54] S. Ghorbani, A. Etemad, and N. F. Troje, “Auto-labelling of markers in optical motion capture by permutation learning,” in *CGI*, 2019, pp. 167–178.
- [55] S. Han, B. Liu, R. Wang, Y. Ye, C. D. Twigg, and K. Kin, “Online optical marker-based hand tracking with deep labels,” *ACM Trans. Graph.*, no. 4, p. 166, 2018.
- [56] S. Ghorbani, “Exploiting novel deep learning architecture in character animation pipelines,” Ph.D. dissertation, York University, 2022, accessed: 2024-09-05. [Online]. Available: <http://hdl.handle.net/10315/40598>

- [57] OptiTrack, “Data editing - optitrack motive documentation,” 2024. [Online]. Available: <https://docs.optitrack.com/motive/data-editing#gaps>
- [58] N. Mahmood, N. Ghorbani, N. F. Troje, G. Pons-Moll, and M. J. Black, “AMASS: archive of motion capture as surface shapes,” in *ICCV*, 2019, pp. 5441–5450.
- [59] M. Loper, N. Mahmood, and M. J. Black, “Mosh: motion and shape capture from sparse markers,” *ACM Trans. Graph.*, no. 6, pp. 220:1–220:13, 2014.
- [60] D. Holden, “Robust solving of optical motion capture data by denoising,” *ACM Trans. Graph.*, no. 4, p. 165, 2018.
- [61] Tobii, “Tobii Pro Glasses 3 - Wearable Eye Tracker,” <https://www.tobii.com/products/eye-trackers/wearables/tobii-pro-glasses-3>, 2024, accessed: September 25, 2022.
- [62] Tobii Pro, *Tobii Pro Glasses 3 Developer Guide*, v1.7 ed., <https://developer.tobiipro.com/tobii-pro-glasses-3.html>, Tobii AB, April 2023, accessed: Sep 21, 2022.
- [63] Tobii, “Head Unit Coordinate System (HUCS),” https://connect.tobii.com/s/article/Head-Unit-Coordinate-System-HUCS?language=en_US, May 2023, accessed: September 25, 2022.
- [64] Y. Pan, R. Agrawal, and K. Singh, “S3: speech, script and scene driven head and eye animation,” *ACM Trans. Graph.*, no. 4, pp. 47:1–47:12, 2024.
- [65] Tobii AB, *Motion Capture Marker Set Product Description*, August 2022, all rights reserved. Copyright © Tobii AB. [Online]. Available: <https://go.tobii.com/motion-capture-product-description>
- [66] J. Kulozik and N. Jarrassé, “Evaluating the precision of the HTC VIVE ultimate tracker with robotic and human movements under varied environmental conditions,” *CoRR*, 2024.
- [67] K. Shoemake, “Animating rotation with quaternion curves,” in *SIGGRAPH*, 1985, pp. 245–254.
- [68] E. Foxlin, “Inertial head-tracker sensor fusion by a complimentary separate-bias kalman filter,” in *VRAIS*, 1996, pp. 185–194.

- [69] A. Harlley and A. Zisserman, *Multiple view geometry in computer vision (2. ed.)*. Cambridge University Press, 2006.
- [70] M. Garau, M. Slater, S. Bee, and M. A. Sasse, “The impact of eye gaze on communication using humanoid avatars,” in *CHI*, 2001, pp. 309–316.
- [71] Y. Ohshima and A. Nakazawa, “Eye contact detection from third person video,” in *ACPR*, 2019, pp. 667–677.
- [72] G. Tevet, S. Raab, B. Gordon, Y. Shafir, D. Cohen-or, and A. H. Bermano, “Human motion diffusion model,” in *ICLR*, 2023.
- [73] M. Zaheer, S. Kottur, S. Ravanbakhsh, B. Póczos, R. Salakhutdinov, and A. J. Smola, “Deep sets,” *CoRR*, 2017.
- [74] H. Maron, H. Ben-Hamu, N. Shamir, and Y. Lipman, “Invariant and equivariant graph networks,” in *ICLR*, 2019.
- [75] J. Wang, C. Rupprecht, and D. Novotný, “Posediffusion: Solving pose estimation via diffusion-aided bundle adjustment,” in *ICCV*, 2023, pp. 9739–9749.
- [76] A. Pepe, J. Lasenby, and P. Chacón, “Learning rotations,” *Math. Methods Appl. Sci.*, 2024.
- [77] Q. Yan, Z. Liang, Y. Song, R. Liao, and L. Wang, “Swingnn: Rethinking permutation invariance in diffusion models for graph generation,” *Trans. Mach. Learn. Res.*, 2024.
- [78] J. Ho, T. Salimans, A. A. Gritsenko, W. Chan, M. Norouzi, and D. J. Fleet, “Video diffusion models,” in *NeurIPS*, 2022.
- [79] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, “Attention is all you need,” in *NeurIPS*, 2017, pp. 5998–6008.
- [80] I. T. Hooge, D. C. Niehorster, M. Nyström, R. Andersson, and R. S. Hessels, “Fixation classification: How to merge and select fixation candidates,” *Behav. Res. Methods*, no. 6, pp. 2765–2776, 2022.
- [81] R. S. Hessels, D. C. Niehorster, C. Kemner, and I. T. Hooge, “Noise-robust fixation detection in eye movement data: Identification by two-means clustering (i2mc),” *Behav. Res. methods*, pp. 1802–1823, 2017.

- [82] S. D. König and E. A. Buffalo, “A nonparametric method for detecting fixations and saccades using cluster analysis: Removing the need for arbitrary thresholds,” *J. Neurosci. Methods*, pp. 121–131, 2014.
- [83] M. Land and B. Tatler, *Looking and Acting: Vision and Eye Movements in Natural Behaviour*. Oxford University Press, 2009.
- [84] A. L. Yarbus, *Eye movements and vision*, 2013.
- [85] I. Fuchs, U. Ansorge, C. Redies, and H. Leder, “Salience in paintings: Bottom-up influences on eye fixations,” *Cogn. Comput.*, pp. 25–36, 2011.
- [86] G. Gredebäck and T. Falck-Ytter, “Eye movements during action observation,” *Perspect. Psychol. Sci.*, no. 5, pp. 591–598, 2015.
- [87] Y. Rubner, C. Tomasi, and L. J. Guibas, “The earth mover’s distance as a metric for image retrieval,” *Int. J. Comput. Vis.*, no. 2, pp. 99–121, 2000.
- [88] E. G. Freedman, “Coordination of the eyes and head during visual orienting,” *Exp. Brain Res.*, pp. 369–387, 2008.
- [89] R. Ackerley and G. R. Barnes, “The interaction of visual, vestibular and extra-retinal mechanisms in the control of head and gaze during head-free pursuit,” *Physiol. J.*, no. 7, pp. 1627–1642, 2011.
- [90] M. Zhu and S. Gupta, “To prune, or not to prune: exploring the efficacy of pruning for model compression,” *arXiv preprint arXiv:1710.01878*, 2017.
- [91] R. Krishnamoorthi, “Quantizing deep convolutional networks for efficient inference: A whitepaper,” *arXiv preprint arXiv:1806.08342*, 2018.
- [92] J. Gou, B. Yu, S. J. Maybank, and D. Tao, “Knowledge distillation: A survey,” *International Journal of Computer Vision*, no. 6, pp. 1789–1819, 2021.
- [93] J. Song, C. Meng, and S. Ermon, “Denoising diffusion implicit models,” *CoRR*, 2020.
- [94] A. F. T. Martins, M. V. Treviso, A. Farinhas, V. Niculae, M. A. T. Figueiredo, and P. M. Q. Aguiar, “Sparse and continuous attention mechanisms,” *CoRR*, 2020.
- [95] Z. Zhang, “A flexible new technique for camera calibration,” *IEEE Trans. Pattern Anal. Mach. Intell.*, no. 11, pp. 1330–1334, 2000.

- [96] M. A. Fischler and R. C. Bolles, “Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography,” *Commun. ACM*, no. 6, pp. 381–395, 1981.

Appendix A

Appendix

A.1 Demographics of Participants

In this study, a total of 74 participants were initially recruited. However, only 48 participants were included in the final dataset used for training and analysis. The exclusion of 26 participants was primarily due to data quality issues caused by high prescription values, which led to missing data during eye-tracking. To ensure the reliability of our findings, only high-quality data were retained for analysis.

A.1.1 Age Distribution

The age range of the selected participants was diverse, spanning from 18 to 40 years, with an average age of 25. Figure A.1 illustrates the age distribution of the 48 included participants.

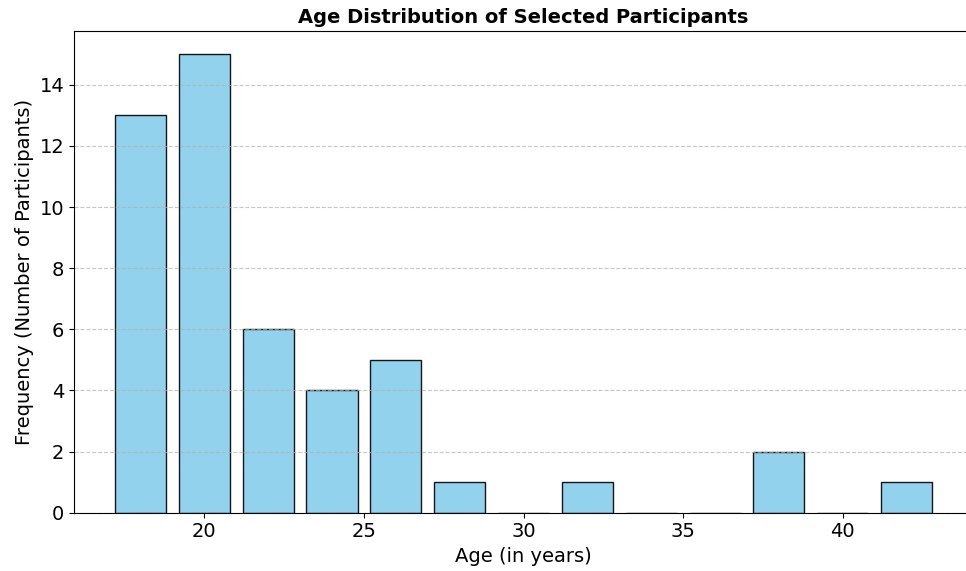


Figure A.1: Age distribution of the 48 participants included in the study. The plot shows a broad range of ages, ensuring representation across different age groups.

A.1.2 Gender Distribution

Of the 48 participants, 62.5% identified as female, while 37.5% identified as male. Figure A.2 presents the gender distribution.

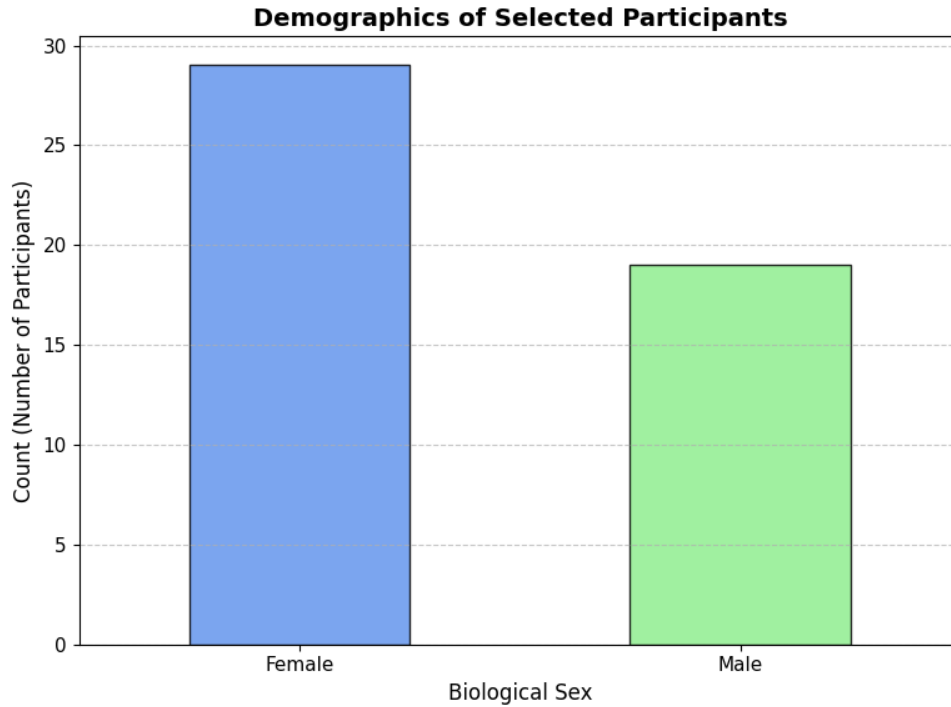


Figure A.2: Gender distribution of the 48 participants included in the study.

A.1.3 Visual Acuity and Prescription Needs

Among the 48 participants, 68% had naturally normal vision, while 32% required vision correction. For participants requiring prescription glasses, the average prescription value ranged from -1.5 to 1.5 . Tobii eye trackers come equipped with a set of lenses, and we tested these participants with the appropriate lenses. Figure A.3 shows the distribution of participants based on visual acuity.

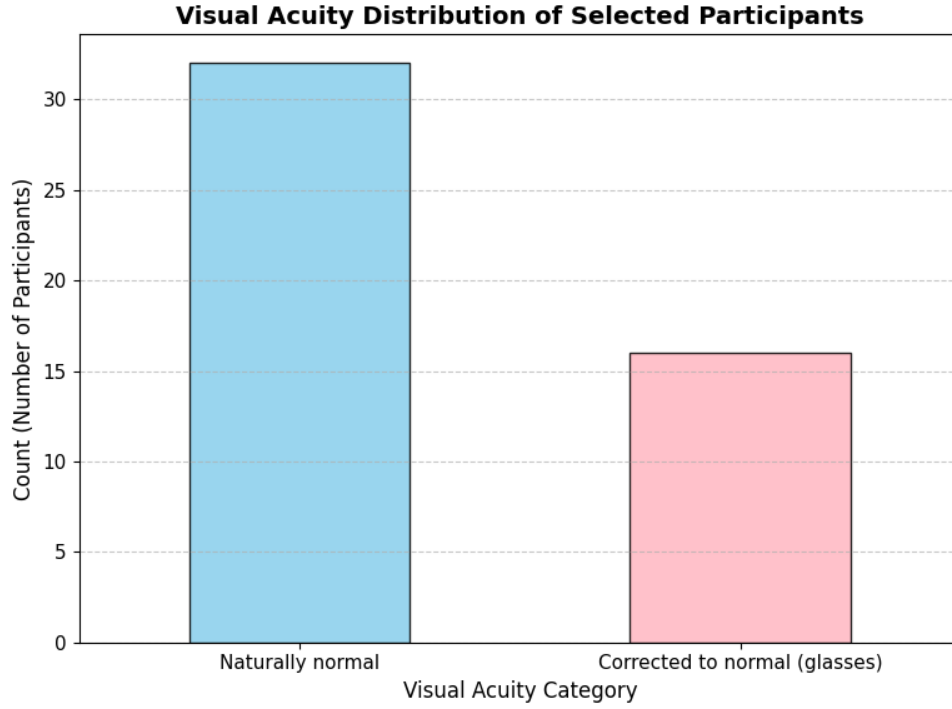


Figure A.3: Distribution of visual acuity among participants, showing the proportion of participants with naturally normal vision and those requiring glasses.

The Tobii eye-tracking system was equipped with the required lenses based on different prescriptions to accommodate participants needing vision correction. This ensured that all participants, regardless of their visual needs, could easily engage with the equipment.

The demographics of the selected participants provided a diverse and representative sample for this study. By ensuring high-quality data and addressing participants' individual needs, we enhanced the reliability and validity of the collected dataset.

A.2 Visualization Tool for Collected Data

To ensure the accuracy and consistency of the collected data, we developed an interactive web-based visualization tool using JavaScript, three.js, and HTML. This tool was designed to serve as a sanity-check mechanism by visualizing the data in 3D, enabling verification of synchronization and alignment across multiple data sources. Specifically, it allowed us to:

- Visualize the 3D spatial data to verify correct transformations into a unified coordinate system.

- Display the recorded videos from the eye-tracker cameras alongside the 3D data to cross-check synchronization.
- Render the planes representing each participant’s face and their gaze directions, aiding the validation of labeled eye-contact moments.

The interactive features of the tool included functionalities to play, pause, and navigate through recordings, which facilitated precise inspection of specific frames. This visualization was critical for confirming the correctness of the data annotations and ensuring high-quality dataset integrity.

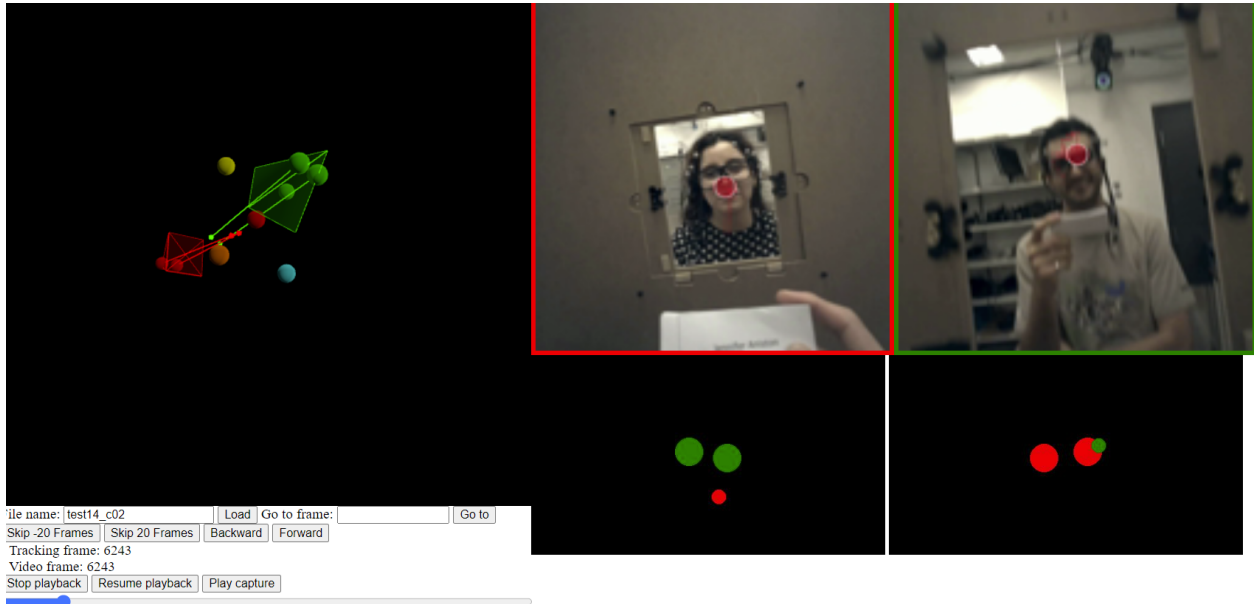


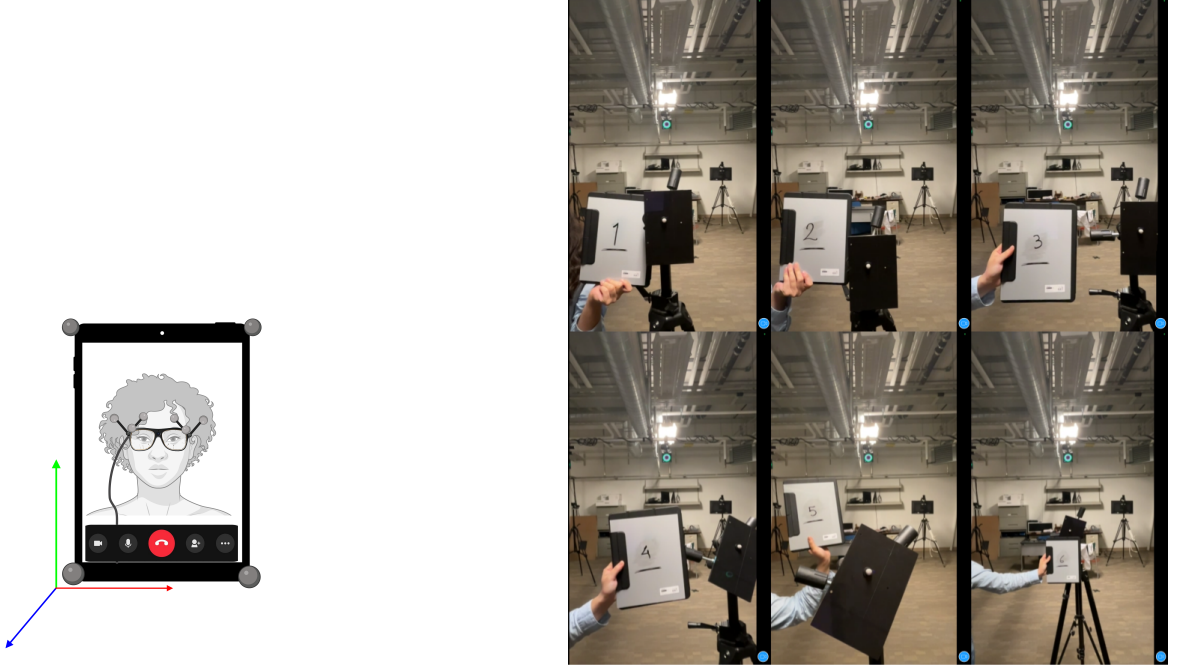
Figure A.4: Web-based visualization tool showing 3D data, participant face planes, and videos from eye trackers. The tool supports interactive navigation through the recordings to play back recorded data and validate synchronization and labeled eye-contact moments.

A.3 Calibration of Zoom Camera and Screen Mapping

To accurately determine where participants were looking on the screen during the Zoom condition, we performed a detailed camera calibration process. This calibration was essential for mapping 2D screen coordinates to our 3D coordinate system.

To calibrate the Zoom camera, we performed a calibration process using a set of images taken with both the Zoom camera (shown in Figure A.5) and a motion capture system. The

motion capture camera was mounted on a tripod in a fixed position relative to the Zoom camera to ensure consistency. We captured 16 images of a calibration object with markers and manually labeled the 2D pixel positions of these markers in the images. The corresponding 3D positions of the markers were known from the motion capture system, which provided precise real-world coordinates.



(a) Zoom coordinate system and markers around iPad screen.

(b) Sample data used for camera calibration.

Figure A.5: Illustrations of the Zoom camera calibration setup. (a) shows the coordinate system and the placement of markers around the iPad screen. (b) displays a sample of the data used for calibrating the camera, highlighting the 2D marker positions and their correspondence with known 3D locations.

The calibration was conducted using the Direct Linear Transformation (DLT) method [69, 95], combined with RANSAC (Random Sample Consensus) [96] to improve robustness. The main steps involved are as follows:

We collected the 3D coordinates of the markers from the motion capture system and manually labeled the corresponding 2D coordinates in the images captured by the Zoom camera. This provided pairs of 3D-2D data points that were used to estimate the camera's projection matrix, which maps 3D world coordinates to 2D image coordinates.

The DLT algorithm was employed to estimate this projection matrix. For each pair of 3D and 2D points, the DLT method constructs a set of linear equations that relate the 3D coordinates to their corresponding 2D positions. These equations were combined into a larger system, which was then solved using Singular Value Decomposition (SVD). The result is a transformation vector that represents the camera’s projection matrix in a 3×4 form, capturing both the intrinsic and extrinsic parameters of the camera.

To assess the accuracy of the estimated projection matrix, we calculated the error [69] by comparing the observed 2D positions of the markers with their projected 2D positions as transformed by the matrix. The error was measured as the sum of the squared differences between these points, providing a quantitative evaluation of the calibration’s precision.

To further refine the calibration and handle any potential noise or outliers, we used the RANSAC algorithm. RANSAC repeatedly selected random subsets of the 3D-2D data pairs to compute the DLT projection matrix and evaluated the resulting error. This process was repeated 10,000 times, with the best transformation matrix (the one with the lowest error) selected as the final estimate. This approach helped ensure that the calibration was not overly influenced by any noisy data points or inaccuracies in labeling.

The final transformation matrix obtained through this process represents the optimal mapping from the 3D world coordinates to the 2D image plane of the Zoom camera. The error associated with this matrix was computed to validate the calibration accuracy, confirming that the camera parameters were effectively determined.

Overall, this calibration method utilized the DLT method for initial estimation and RANSAC for refinement to determine the Zoom camera’s parameters, facilitating an approach to zoom camera calibration.