Pomona College
Department of Computer Science

Eyes in Continued Motion: Enhancing Gaze Controlled Systems for Safe Assistive Robotic Interaction

Francisco X. Morales Puente

May 4, 2026



Submitted as part of the senior exercise for the degree of
Bachelor of Arts in Computer Science

Professor Anthony Clark

Copyright © 2026 Francisco X. Morales Puente

The author grants Pomona College the nonexclusive right to make this work available for noncommercial, educational purposes, provided that this copyright statement appears on the reproduced materials and notice is given that the copying is by permission of the author. To disseminate otherwise or to republish requires written permission from the author.

Abstract

I'm developing a low-cost, webcam-based eye-tracking interface to control a wheelchair, enabling intuitive, hands-free mobility for individuals with limited motor function. The system is designed to reduce user fatigue by allowing users to draw paths using their gaze, instead of issuing continuous commands. I also worked on a plane segmentation system that is able to identify walls, floors, and people in order to identify where people are going and whether a path is valid and would not result in collision.

Acknowledgments

Thank you to Professor Clark for being an amazing advisor and supporter, and to Professor Papoutsaki for helping me first think through this project and scope it out. Thank you also to Jack Chin, Ella Zhu, Haram Yoon, and Luis Mendoza for their support with specific parts of testing and iteration. Lastly, I would like to thank my family and friends for their support throughout this research journey.

I would also like to acknowledge the use of AI in the form of Grammarly to revise the text throughout my report to correct any typos, grammatical errors, and awkward phrasing.

Contents

Abstract i

Acknowledgments iii

1 Introduction **1**

2 Related Works **3**

2.1 Gaze Recognition for Focus and Intent 3

2.2 Project Design for Gaze-Based Systems 4

2.3 User Trust 4

2.4 Prior Work vs. My Approach 5

3 Methods **6**

3.1 Gaze Detection and Path Drawing 6

3.2 Obstacle and Human Detection 8

3.3 Integrated System 9

4 Ethics **11**

5 Results **12**

5.1 Gaze Evaluation 12

5.2 Obstacle and Human Evaluation 14

5.3 Overall Interpretation 16

6 Conclusion **17**

Bibliography **18**

Chapter 1

Introduction

Around 12.2% of Americans have some sort of mobility disability, while 7.7% have some disability that limits their ability to live independently [1]. Mobility disabilities can take many different forms, including cerebral palsy, spinal cord injuries, ALS, quadriplegia, and locked-in syndrome. Many of these individuals require lifelong assistance and frequently rely on assistive technologies to regain control over their environment.

A wide range of assistive technologies already exist to help restore agency to disabled individuals: from voice-controlled systems to joystick-controlled wheelchairs. However, these systems still remain limited for users with extremely restricted dexterity.

Eye tracking offers a promising alternative by enabling individuals to navigate and interact with their environment without having to rely on any limb movement. Yet, many eye-tracking systems explored in research depend on expensive or inaccessible hardware such as EOGs, EEGs, or commercial infrared cameras. This is particularly significant because, as it currently stands, around one in four adults with disabilities have an unmet healthcare need due to cost within the past year.

I developed a low-cost system aimed at enabling users to draw paths using their gaze, with the long-term goal of supporting safe movement through actual real world environments. This project covers two main components: mapping gaze to screen coordinates for path drawing, and interpreting the surrounding scene through plane segmentation and human motion estimation. By shifting from continuous teleoperation to path-based control, the system aims to reduce user fatigue while enabling more intuitive interaction.

With this goal in mind, the system was designed for comfort, usability, and consistency rather than optimizing just for quantitative metrics. Design choices such as personalized calibration, our data collection, and gaze smoothing were driven by how the system behaves in practice. Evaluation remained quantitative, using measures like accuracy and drift, but these were interpreted through their impact on tasks such as selecting regions, recognizing direction, and following simple paths. The goal was not perfect precision, but a system that is reliable enough for meaningful real-world use while staying accessible, safe, and intuitive.

Chapter 2

Related Works

Gaze-based control has become instrumental as a modality for robotic control, specifically for human-robot interaction in the shape of assistive technologies or collaborative systems. Recent work has explored understanding user gaze to infer intent, human centered design for a more robust gaze detection system, and developing methods to enhance safety and trust in human-robots systems.

2.1 Gaze Recognition for Focus and Intent

The *Midas Touch Problem* refers to unintentional glances or gazes being misinterpreted as commands. Schirmer et al. [2] developed 3 different models for gaze classification, categorizing eye gaze into fixation, scanning, and task-switching which enabled robots to interpret human gaze cues with greater accuracy to enhance alignment. Their framework emphasizes that gaze must be understood not as just a spatial point but as a behavior over time, providing essential cues for distinguishing intentional control from natural exploration when looking at the world.

Lin et al. [3] extend gaze understanding through GazeHTA, an end-to-end framework that accurately identifies which object a person is attending to by combining pretrained diffusion models, semantic scene features, and explicit head–target association maps. This approach demonstrates how deeper scene awareness can reduce ambiguity when interpreting user focus, something essential for anticipating the movements of other individuals

in real world environments. Similarly, Arreghini et al. [4] present a ROS2 based system capable of detecting mutual gaze at distances up to five meters, highlighting long-range eye contact as a valuable cue for inferring user intent. Saran et al. [5] similarly explore a deep learning model that predicts both referential and mutual gaze in real time, leveraging contextual information to improve the robot's ability to anticipate user goals. Together, these methods show how recognizing where and why a user is looking can significantly enhance the fidelity and reliability of gaze-controlled systems.

2.2 Project Design for Gaze-Based Systems

Designing effective gaze-based robotic systems requires the purposeful selection of tracking hardware, calibration procedures, and a user interface layout. Tokmurziyev et al. [6] demonstrated a wearable gaze-controlled manipulation system in which an ESP32 mounted eye tracker and a robot-mounted camera work together to provide stable gaze estimation and object selection. They also use a 35 point calibration procedure which highlights the importance of robust gaze alignment, especially when the user interacts through a live video feed. Lee et al. [7] similarly emphasize interface design through an augmented-reality control environment in which users guide a robot by fixating on virtual overlays mapped onto the real world. Their results show that a thoughtful and purposeful UI configuration can make gaze commands as effective and intuitive as joystick control.

Moreover, several works explore simplified screen based layouts for telemanipulation based gaze control. For example, Minamoto et al. [8] implemented a 3x3 grid overlay on a camera feed, allowing the robot to move in the direction corresponding to the user's gaze position and stop when the user looked at the center. This grid driven interface offers a minimally straining method directly relevant to wheelchair or navigation control.

2.3 User Trust

Human-robot systems require a multi-faceted array of considerations to ensure they are safe and trustworthy. Tabatabaei et al. [9] explores how human gaze behavior can be used to detect robot errors during cooperative activities by analyzing gaze patterns. The ability

for a robot to recognize when the user believes it has made an error is instrumental for trust, as it enables the robot to react, recover, or clarify rather than continuing an unsafe or unintended action. Likewise, Saran et al. [5] highlights the safety benefits of accurately predicting both mutual and referential gaze in real time, allowing a system to distinguish deliberate commands from incidental attention. Together these two show, a system that can infer user confusion or concern through gaze signals provides an additional safety layer especially valuable in assistive scenarios where errors can have increased physical consequences from mistakes.

Beyond just error detection, trust is built by intuitive and dependable controls. Dominijanni et al. [10] demonstrates how a multi modal interface can reduce the cognitive load and promote consistent performance from users without explicit feedback on the control system. Moreover, Edughele et al. [11] further emphasizes that robust calibration and a fine tuned system enables improved communication and utility between robots and their users.

2.4 Prior Work vs. My Approach

Prior work has advanced gaze estimation, intent prediction, and safe interactions. However, most existing systems depend on specialized hardware such as infrared eye trackers or expensive setups that are inaccessible to the general public. My approach, by contrast, emphasizes low cost by relying solely on a standard webcam, prioritizing affordability.

Additionally, existing systems typically focus on either gaze-driven intent recognition or telemanipulation. In contrast, I develop a system that allows users to draw paths with their gaze, reducing the need for continuous control and lowering user fatigue. Unlike much of the related work, this project is centered on the human experience of using the system. It is designed with the primary user in mind, placing greater emphasis on affordability and qualitative measures such as comfort, intuitiveness, and usability alongside quantitative performance.

Chapter 3

Methods

This project combines two real-time computer vision pipelines into a single assistive interaction system. One pipeline estimates gaze and maps it to screen intent. The other segments the scene into floor and wall regions and adds lightweight human tracking and motion cues. Throughout development, I prioritized methods that were fast, simple, and reliable enough for live interaction.

3.1 Gaze Detection and Path Drawing

I developed the gaze system in two stages; however, the first stage was just an early prototype. It used a dlib-based approach with facial landmarks, eye masks, simple geometry, and directional heuristics. From the detected eye regions, I estimated the general gaze of a user by looking at relative eye shape, darker pixel distributions inside the eye, and changes in horizontal and vertical position. This was an improved version of my earlier work, *Eyes in Motion*, which also relied on pixel-based eye analysis.

When I tested this version for path drawing, its limitations became apparent very quickly. It struggled to reduce fatigue and did not allow for intuitive drawn paths. It was also very sensitive to lighting and occlusion, and the feature estimates became too tied to my specific facial structure. While it was useful as a starting point, it did not hold up well enough to become the main system.

The main system that followed was much more centered on personalized learning.

Instead of relying on overfit heuristics, I moved to a MediaPipe-based pipeline paired with a small custom CNN model. MediaPipe gave me a denser and more stable facial and iris landmarks, which made it possible to build a better input representation. From this, I used two types of input: a grayscale crop of the eye region and a small set of geometric features describing iris position, eye openness, and head orientation. These were fed into my custom model, *TinyHybridScreenGazeNet*, which processes the eye image and geometric features separately before combining them. The model predicted continuous screen coordinates, a coarse gaze direction, and whether the current frame was valid.

A large part of the work was not just the model itself, but building the collection and training loop around it. Before collection, I used a seating guide to record how the user was positioned relative to the screen and camera so that training and live use stayed consistent. After that, the user followed targets across the screen to collect examples from the corners, edges, and interior, and I also collected invalid pictures like when the user is mid blink, squinting, or partially closing their eyes. This made my model a lot more user-specific. In practice, this collection process only took a few minutes (around 5), but it was something that I iterated upon over and over trying to improve how we collected data to ensure an accurate and precise model customized to the user for their comfort.

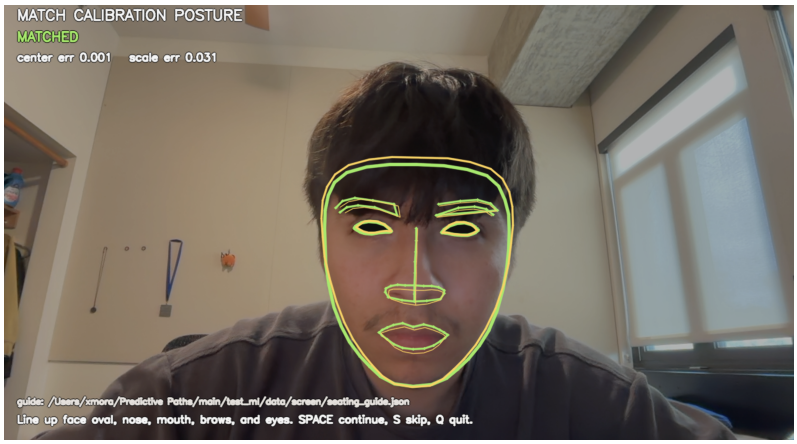


Figure 3.1: Seating guide used to align the user before data collection and live interaction.

Moreover, training was iterative. After data collection, I would train the initial model, evaluate where it was weak, collect more samples in for those specific regions, and retrain. That was especially important for the harder parts of the screen, since weak regions were

usually not evenly distributed. During evaluation, I also noticed that when looking around the screen, the cursor would jump a lot, since our eyes moved faster than the camera could reliably capture.

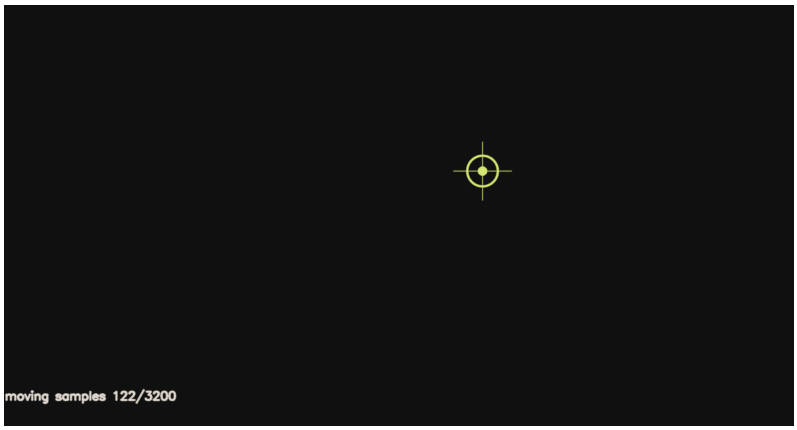


Figure 3.2: Data collection interface used to gather personalized gaze samples across the screen.

To deal with this, I added smoothing at inference so that the output behaved more like a stable cursor than a sharp singular point that jumped around. Thus, instead of jumping directly to each predicted location, the cursor moves gradually toward it, and frames marked as invalid are dampened to avoid sudden jumps. The goal wasn't just to predict gaze, but to make the output actually usable.

Finally, with a stable cursor in place, I used it to enable simple path drawing, where users could start and stop drawing using a specific blink pattern. This introduced a concrete way to evaluate the system and establish a working interaction loop. While the paths themselves were still limited, this stage gave me something concrete to build on as I moved into environment segmentation and human detection.

3.2 Obstacle and Human Detection

Path drawing only becomes useful for assistive movement if the system can understand what the path is moving through. Hence, the obstacle pipeline provides for a real-time understanding of the environment by identifying floors, walls, and people. Instead of trying to reconstruct the full 3D structure of the scene, I used SegFormer-B0 to label each

pixel as part of the floor, wall, person, or another region. I then grouped these labels into broader categories so the output stayed simple and easy to interpret. To keep the system responsive, I focused on the newest camera frame instead of letting frames build up in a queue. This helped keep the segmentation overlay matched to what the user was seeing in real time.

I also slightly expanded wall boundaries in the segmentation output to prevent nearby openings from being misinterpreted as free space.

Building on this, human detection was handled directly using the same segmentation output. Instead of introducing a separate heavy model, I used the person regions already identified in the scene and stabilized them over time so they would not flicker or jump between frames. When these regions became unreliable, I used a lightweight (HOG) fallback detector to maintain consistent tracking.

Lastly, to estimate movement, I tracked how each detected person shifted across frames and smoothed that motion so the direction would not change suddenly because of noise. When available, I also used rough face and body landmarks to better estimate which way a person was facing or likely to move. This gave the system a simple way to identify when someone was present and understand their general movement direction, without needing a complex pose or intent prediction model.

The goal was not perfect recognition, but to provide stable and readable cues that could be used alongside gaze input.

3.3 Integrated System

The final system combined both pipelines into a single live interface. The camera view acted as the main display, with gaze input overlaid as a cursor or path, along with floor and wall regions and human motion cues. The intended use case is a user with limited mobility interacting through eye movement, where the user draws a path with their gaze while the system provides context about the surrounding scene.

The system does not yet perform full navigation planning, and the gaze-drawn path is still too unstable to be used reliably. While the goal was path-based control, this



Figure 3.3: Integrated system combining gaze input with scene understanding and human motion cues.

component is still in progress. For now, the system is better suited for interpreting general intent rather than precise paths.

Chapter 4

Ethics

This project works toward a low-cost gaze-based system that could eventually support wheelchair navigation by letting users draw paths and understand the scene around them. However, in its current form, the system should only be seen as a prototype. As it does not yet focus on full wheelchair control.

The first ethical concern is privacy. The system relies on biometric information from the user's face and iris landmarks. Users should know what data is collected, why it is needed, how it is stored, and whether it is deleted or reused after training.

Safety and autonomy are also central. Gaze-based systems can misread user intent, especially during blinks, natural looking behavior, or unstable predictions. Any future physical deployment would need clear ways for the user to pause, cancel, or override the system so control remains with the user.

Accessibility is equally important. Since this project is motivated by assistive use, it should remain affordable, work across different users and environments, and avoid relying on expensive hardware. Ethical design here means treating the current system as a foundation, while prioritizing user dignity, low frustration, and user-centered control.

Chapter 5

Results

I evaluated my system through live testing across different tasks in both the gaze system and the obstacle and human detection pipeline, focusing on how well it works in practice and where it still struggles.

The evaluation was split into two parts. The first measured gaze performance: whether my system could identify where a user was looking, distinguish broad directions such as left and right, how evenly it covered the screen, and how it handled continuous tasks like tracing a shape. The second measured the obstacle and human pipeline: whether my system maintained consistent wall and floor identification, reliably tracked a person, and correctly recognized that person's movement direction.

5.1 Gaze Evaluation

The first test I ran was a target selection test designed to measure whether my system could correctly identify the general region of the screen that a user was looking at. I displayed targets in a simple 3x3 grid and asked the user to look at each region. In this test, my system correctly identified the intended screen region **82.89%** of the time across **135 trials**. This showed that when a user looked at a general area of the screen, my system usually placed the prediction in that same region. However, the performance was not uniform across the screen. The system was less reliable in certain areas, especially along the bottom of the screen, where predictions were less consistent.

This uneven behavior is shown in Figure 5.1.



Figure 5.1: The 3x3 test shows region selection accuracy

Next, I ran a screen coverage test using a 5x3 grid. This test was meant to evaluate whether my system could reliably recognize gaze across the full screen rather than only in a few easier regions. This turned out to be one of the main weaknesses of the system. Only **26.7%** of regions met the reliability threshold, and several areas were rarely recognized correctly. This showed that the problem was not simply selecting a region once, but maintaining consistent performance across the entire screen as seen in Figure 5.2.

Next, I ran a five-direction test where I was prompted to look at one of five different directions within a certain time frame: forward, left, right, up, and down. In this test, my system achieved **75.0%** accuracy while producing valid predictions in **100.0%** of trials. The errors were not random. Left was often interpreted as up, and up was often interpreted as right. This suggested that my system was able to capture general direction, but still struggled to separate similar gaze angles reliably.

Finally, I tested continuous gaze control through a shape tracing task. In this test, the user attempted to follow simple shapes using only eye movement. This was the most difficult task for the system. It was only able to loosely follow the intended paths, with limited overlap and coverage of the target shapes. Figure 5.3 shows examples of these traces compared to the intended shapes.

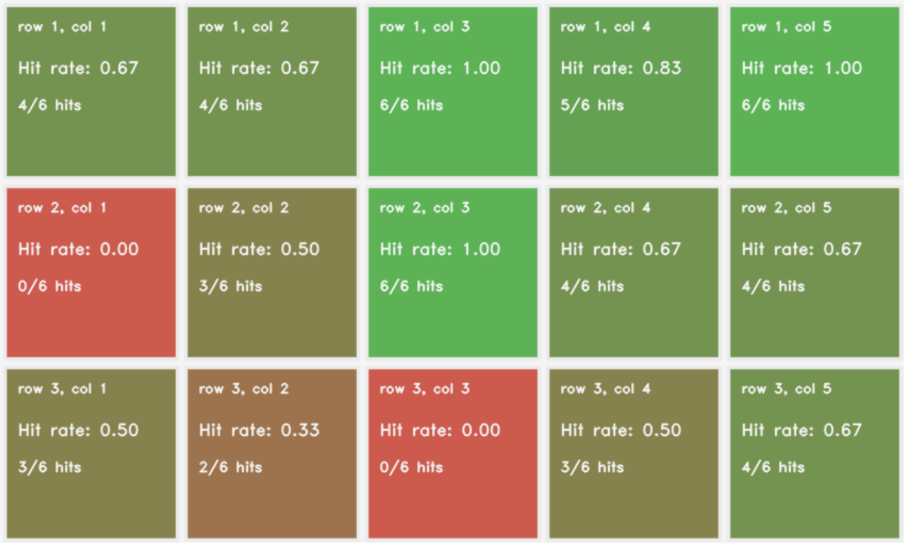


Figure 5.2: The 5x3 test shows how consistently different parts of the screen could be tracked.

Overall, these results showed that my system already supported coarse interaction reasonably well, especially for region selection and basic directional input. However, it became less reliable when the task required full-screen consistency or fine continuous control.

5.2 Obstacle and Human Evaluation

I evaluated the obstacle and human pipeline through three live tests: a room scan, a person identifying test, and a movement direction test.

During the room scan, I moved the camera around the environment and observed whether my system continued to detect floor and wall regions as the view changed. My system detected floor and wall structure across **96.0%** of frames, and these predictions remained stable as the camera moved, with floor regions showing slightly higher consistency than walls. This showed me that the system maintained a reliable understanding of the general room layout, although this test measured stability rather than accuracy.

In the person identifying test, I stood in front of the camera without leaving the frame and measured whether my system could keep me tracked over time. My system maintained tracking for **100.0%** of the trial, keeping the person continuously in view for

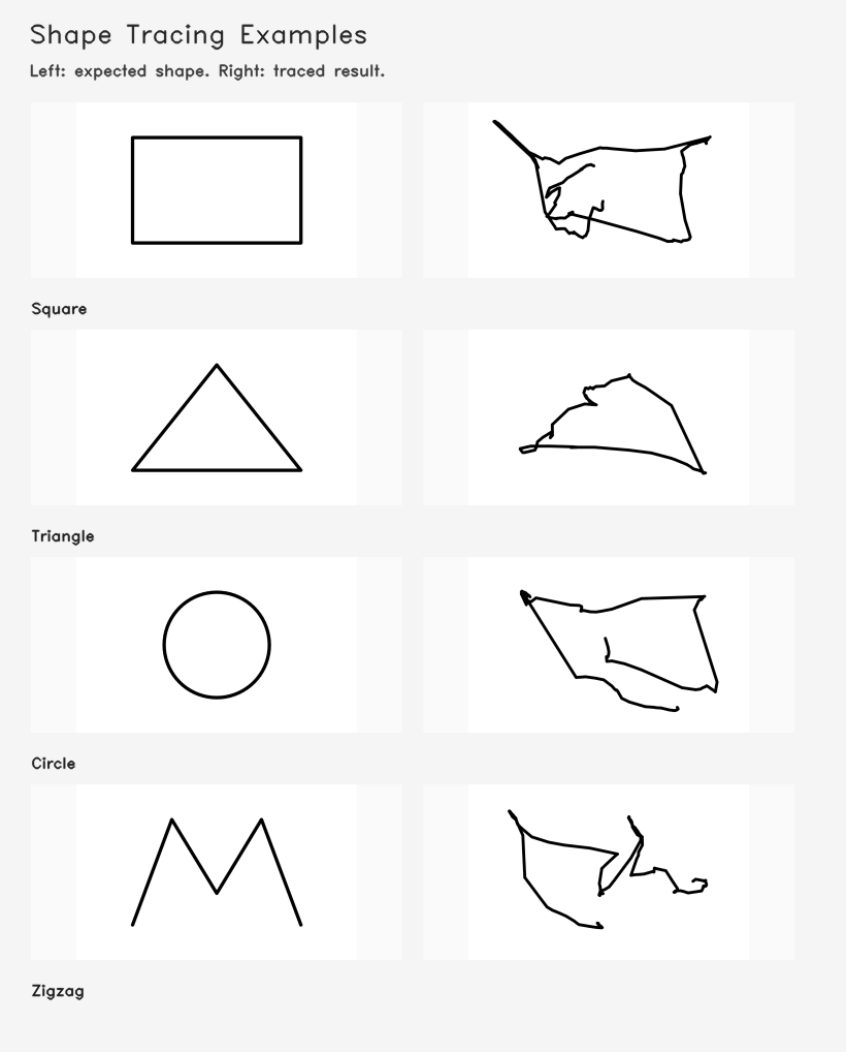


Figure 5.3: Examples of gaze-based shape tracing. The left images show the target shapes, and the right images show the paths produced by my system.

nearly the entire duration. This showed that once a person was detected, the system could reliably maintain that detection.

In the movement direction test, I moved in four directions: left, right, up, and down. The goal was to measure whether my system could correctly recognize the direction of motion while tracking the person. In this test, my system achieved **100.0%** overall direction accuracy, meaning it correctly identified the intended direction in each trial. The average frame-level direction match rate was **78.8%**, which showed that some directions were tracked more cleanly than others over time. Overall, this showed that my system could reliably track a person and recognize broad movement direction.

5.3 Overall Interpretation

My system worked best when the task was simple and stable. For gaze, this meant selecting general areas of the screen and recognizing basic directions. For the obstacle pipeline, it meant keeping a consistent sense of the scene and reliably tracking a person. However, it struggled more when the task required precision or consistency across the entire screen. This showed up in uneven coverage, difficulty with tracing, and unreliable movement direction recognition.

Overall, my system already supports basic interaction well, but it is not yet consistent enough for more precise control.

Chapter 6

Conclusion

This project explored how a low-cost, webcam-based system could use gaze as a practical input for interaction. The results show that gaze can already support coarse tasks like region selection and basic directional input, and that combining it with simple scene understanding creates a useful foundation for assistive interaction. At the same time, the system is not yet reliable for precise control. Performance drops across the full screen, and continuous tasks like tracing or path drawing remain unstable.

Overall, this work shows that gaze-based control is promising, but still requires refinement to be fully usable in real-world settings. Future work should focus on improving stability, expanding reliable screen coverage, and making path-based interaction more consistent. Just as importantly, this work shows that meaningful assistive interaction does not have to rely on expensive or specialized hardware. Even in its current form, the system demonstrates that simple, accessible tools can support more affordable approaches to assistive technology.

Bibliography

[1] Centers for Disease Control and Prevention. Disability impacts all of us. `https://www.cdc.gov/disability-and-health/articles-documents/disability-impacts-all-of-us-infographic.html`, 2024. Accessed: 2025-11-16.

[2] Fabian Schirmer, Philipp Kranz, Chad G. Rose, Volker Willert, Jan Schmitt, and Tobias Kaupp. Utilizing eye gaze for human-robot collaborative assembly. In *2025 20th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 1603–1607, 2025.

[3] Zhi-Yi Lin, Jouh Yeong Chew, Jan van Gemert, and Xucong Zhang. Gazehta: End-to-end gaze target detection with head-target association. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9447–9454, 2025.

[4] Simone Arreghini, Gabriele Abbate, Alessandro Giusti, and Antonio Paolillo. A long-range mutual gaze detector for hri. In *Proceedings of the 2024 ACM/IEEE International Conference on Human-Robot Interaction*, HRI '24, page 870–874, New York, NY, USA, 2024. Association for Computing Machinery.

[5] Akanksha Saran, Srinjoy Majumdar, Elaine Schaertl Short, Andrea Thomaz, and Scott Niekum. Human gaze following for human-robot interaction. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 8615–8621, 2018.

[6] Issatay Tokmurziyev, Miguel Altamirano Cabrera, Luis Moreno, Muhammad Haris Khan, and Dzmitry Tsetserukou. Gazegrasp: Dnn-driven robotic grasping with

wearable eye-gaze interface. In *2025 20th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 1680–1684, 2025.

[7] Joosun Lee, Taeyhang Lim, and Wansoo Kim. Investigating the usability of collaborative robot control through hands-free operation using eye gaze and augmented reality. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4101–4106, 2023.

[8] Masahiko Minamoto, Yutaro Suzuki, Takahiro Kanno, and Kenji Kawashima. Effect of robot operation by a camera with the eye tracking control. In *2017 IEEE International Conference on Mechatronics and Automation (ICMA)*, pages 1983–1988, 2017.

[9] Ramtin Tabatabaei, Vassilis Kostakos, and Wafa Johal. Real-time detection of robot failures using gaze dynamics in collaborative tasks. In *2025 20th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 1660–1664, 2025.

[10] Giulia Dominijanni, Daniel Leal Pinheiro, Leonardo Pollina, Bastien Orset, Martina Gini, Eugenio Anselmino, Camilla Pierella, Jérémy Olivier, Solaiman Shokur, and Silvestro Micera. Human motor augmentation with an extra robotic arm without functional interference. *Science Robotics*, 8(85):eadh1438, 2023.

[11] Hilary O. Edughele, Yinghui Zhang, Firdaus Muhammad-Sukki, Quoc-Tuan Vien, Haley Morris-Cafiero, and Michael Opoku Agyeman. Eye-tracking assistive technologies for individuals with amyotrophic lateral sclerosis. *IEEE Access*, 10:41952–41972, 2022.

[12] M. S. H. Sunny, M. I. I. Zarif, I. Rulik, et al. Eye-gaze control of a wheelchair mounted 6dof assistive robot for activities of daily living. *Journal of NeuroEngineering and Rehabilitation*, 18(1):173, 2021.