A Comprehensive Evaluation of User Experience in Eye-Controlled Interaction

Hanwen Zhang
Glasgow College, University of Electronic Science and Technology of China
Department of Electronic Information Engineering
Chengdu, Sichuan, 611731, China
Email: 2023190503038@std.uestc.edu.cn

Abstract

Eye-controlled interfaces offer a hands-free and intuitive means of human-computer interaction, increasingly used in healthcare, gaming, and assistive technologies. Yet, challenges such as low precision, unintentional activation (e.g., the "Midas Touch" effect), and visual fatigue persist. This paper presents a comprehensive evaluation of user experience in eye-controlled systems, with an emphasis on intent recognition accuracy, system feedback, and fatigue mitigation. A novel contribution of this work is the integration of machine learning algorithms, such as recurrent and convolutional neural networks, for modeling gaze trajectories and predicting user intent. A simulated experiment is conducted to assess the performance of the proposed models in reducing unintended activation. Real-time optimization strategies, including model compression and edge deployment, are also discussed. The findings suggest that ML-enhanced gaze interaction can improve responsiveness, accuracy, and user satisfaction, providing a promising path toward robust, fatigue-aware, and personalized gaze-based interfaces.

Keywords: Eye-controlled interaction; User experience; Intent recognition; Feedback design; Visual fatigue

1. Introduction

Eye-based interaction leverages ocular motion parameters—such as fixation points, saccadic paths, and pupil dynamics—as input signals, offering intuitive, hands-free, and high-degree-of-freedom interaction particularly suitable for individuals with mobility limitations or for use in constrained environments [1][2]. With ongoing advancements in tracking accuracy and system latency, gaze-based interaction is transitioning from laboratory settings to real-world applications, including assistive reading for the visually impaired, eye-controlled wheelchairs, AR menu selection, and VR target acquisition.

Although eye-based interaction is common and convenient, it also brings some problems in user experience. There are many unintentional signals in our eye movements, such as tiny quick rotation, looking back and scanning. The system often mistakenly takes these as the commands we want to input, which leads to the famous "Midas Touch" problem [3]. Also, in the process of interaction, if there is not enough feedback and confirmation mechanism, it is easier for users to feel nervous, make mistakes or make some unclear actions. Using this technology for a long time may also make eyes tired and dizzy, and increase the burden on our brains [4][5].

From the point of view of our users, this paper systematically evaluates the important problems that affect those systems that operate with eyes. It refers to the previous research techniques and how to make people more comfortable to use. The evaluation mainly depends on three aspects: (1) the system recognizes whether our intention is accurate or not, (2) the system is opaque, whether feedback is given or not, and (3) whether it will be tired after being used for a long time and whether it can be used for a long time. Based on an organized literature review and case comparison, we put forward some practical design strategies, hoping to help make the next generation of high-performance gaze-based system.

From the computer point of view, it is really difficult to accurately distinguish between intentional blinking and natural blinking. Those old-fashioned rule systems, such as fixed residence time or simple rules of thumb, are usually not flexible enough to adapt to different users, different tasks and changing environments. Moreover, these systems are difficult to adapt to everyone's eye behavior differences, and often trigger by mistake.

Recently, machine learning(ML) has made great progress, especially those deep learning models that can handle time and space data, which brings a better "intention recognition" method to devices controlled by eyes. For example, Recurrent Neural Networks (RNNs) can learn the time sequence of eye gaze points, while Convolitional Neural Networks (CNNS) can find out the spatial characteristics from heat maps or line-of-sight density maps. With these models, the eye interaction system can adapt to different situations more intelligently, reduce wrong input, and make us feel more natural to use.

This paper contributes to this emerging direction by integrating machine learning into the evaluation framework of gaze-based interaction. Specifically, we present a simulated experiment comparing traditional and ML-based intent recognition strategies in terms of false activation rate. We further discuss deployment considerations for real-time applications, including model compression, edge computing, and personalization through federated learning.

2. Machine Learning for Intent Recognition in Gaze-Based Systems

2.1 Motivation and Model Selection

Accurate intent recognition is central to effective gaze-based interaction. While traditional techniques—such as fixed dwell times or handcrafted heuristics—offer simplicity, they often fail to adapt to the dynamic and personalized nature of gaze behavior. These methods tend to generate high false activation rates, especially in cluttered interfaces or under user fatigue, where involuntary fixations may be misinterpreted as commands.

Machine learning, particularly deep learning, offers a data-driven approach to modeling such complexity. Recurrent Neural Networks (RNNs), including Long Short-Term Memory (LSTM) units, can capture sequential dependencies in fixation data, such as movement velocity, direction, and contextual gaze patterns. Alternatively, Convolutional Neural Networks (CNNs) are adept at identifying spatial structures from gaze heatmaps, which represent the density and clustering of gaze over time.

These models allow for the development of adaptive and context-aware systems that can distinguish between exploratory and intentional gaze behaviors with greater precision.

2.2 Simulated Experiment Design

To validate the effectiveness of ML-based intent recognition, we designed a computational experiment simulating a gaze-based selection interface. The environment included a grid-based virtual layout where targets appeared randomly, and gaze sequences were synthetically generated to mimic user attention, including both intentional and exploratory fixations. Noise patterns and microsaccades were injected based on real-world gaze data statistics to simulate natural eye behavior.

We compared two models: baseline and ML-enhanced. Where baseline is traditional dwell-time-based activation mechanism with a 600ms threshold, and ML-enhanced is an LSTM classifier trained to distinguish between intentional and non-intentional gaze sequences using labeled data. Each model was evaluated using precision, recall, and false positive rate (FPR), with the goal of minimizing unintended activations while maintaining high responsiveness.

2.3 Results and Analysis

Model	False Positive Rate (FPR)	Precision	Recall
Dwell-time	21.3%	78.2%	81.7%
LSTM-based ML	8.9%	90.4%	84.5%

The LSTM model substantially reduced the false activation rate compared to the baseline, demonstrating a 58.2% relative improvement. It also achieved higher precision, indicating fewer unintended commands, and improved recall, suggesting better sensitivity to genuine input.

This experiment highlights the potential of deep learning models to improve gaze-based intent recognition without compromising natural interaction flow.

2.4 Real-Time Deployment Considerations

While ML models show strong offline performance, deploying them in real-time systems introduces computational challenges. Deep networks, especially recurrent architectures, can be resource-intensive, making latency a concern for embedded or mobile platforms.

To address this, we discuss three optimization techniques. The first technique is called “Model Quantization”, which means reducing weight precision (e.g., 32-bit to 8-bit), lowering memory usage and inference time. The second one is called “Pruning and Sparsity”, which means eliminating redundant neurons or connections to simplify the network. The last one is called “Knowledge Distillation”, which means training a smaller “student” model to replicate the performance of a larger, more complex “teacher” model.

In addition, edge computing platforms such as NVIDIA Jetson Nano or Google Coral TPU can run the optimized model very close to users, which can ensure low latency and protect our privacy.

2.5 Towards Personalized Models via Federated Learning

Different users have different habits of seeing things, which is a big problem for the general AI model. Federated learning may be a good way to solve this problem. In this way, the model can be trained on multiple users' devices together, but it is not necessary to share the original eye tracking data, which can be personalized and protect privacy.

Federated learning can slowly adapt the model to everyone's eye movement habits, so that the accuracy will be improved and there is no need for old calibration. This method is especially suitable for barrier-free applications, because personalized response speed is particularly important.

3. System Feedback and Interaction Clarity

When using traditional methods such as touch screen or mouse, the feedback is usually fast and rich-there will be sound, picture change and sometimes vibration when you click. But in the system controlled by eyes, because there is no obvious action, it is difficult for users to know whether the computer has received the instruction or not. This kind of situation without clear feedback can make people feel uncertain, confused and even annoyed, especially when operating complex interfaces.

A common problem is that those systems that can be operated with eyes often suddenly perform operations when we don't know which step it has taken. For example, when using the "gaze selection" function, we may not know whether the system has noticed that we are staring at something or how long it will take for the command to be executed. This kind of uncertainty can make our behavior very strange, for example, we may move our eyes too early, or stare for a long time for fear of not being recognized [3][4].

In order to make the interaction clearer, researchers have come up with many methods of visual feedback. The most common way is to add a progress bar-such as a lap timer, a progress bar or a target that will be reduced-so that you can directly see how long you will stay. These tips allow users to know when it will trigger in advance, adjust their gaze time, and reduce the feeling of anxiety and waiting [6].

Some studies recommend the use of peripheral or line-of-sight feedback design, so that interface elements can make subtle responses according to the movement of our eyes without disturbing our main task at hand. For example, Kiefer et al. [6] put forward a method called foveated visualizations, which can dynamically adjust the clarity of display content or the prominence of key elements according to where our eyes are staring. These designs not only make the response more sensitive, but also save system resources and make the picture look cleaner.

Besides looking at the tips on the screen, we can also use sound to help. For example, when you can't see the screen, a short "beep" or a "ok" voice can let you know that your operation has been accepted by the computer.

With the intention prediction function based on machine learning, the feedback mechanism can become more flexible and personalized. For example, if the system is sure what the user wants to do through gaze trajectory classification, the progress bar can be shortened dynamically or not displayed at all. Conversely, if the eye trajectory is ambiguous, the system will pop up a more conspicuous prompt or ask the user for a second confirmation.

To make the system respond so quickly, it is necessary to make the ML classifier and the feedback layer work closely together to form a closed loop. In this way, the system can not only adjust what feedback to display, but also adjust how and when to display it. This self-adjusting feedback system is expected to greatly increase users' trust in us, and at the same time make it smoother and less brain-consuming.

Despite these improvements, there is still a big problem: giving users too much feedback will distract them. To be clear and simple, you have to design the interface well and test the user's response. The system should provide just enough feedback to let users know what happened, but it should not be too annoying or make them feel tired.

Future research should explore data-driven feedback adaptation, so that the system can learn the most appropriate feedback method from our behavior, task scenes, and physiological signals such as pupil dilation or blink frequency. Such a mechanism can help to realize completely personalized and responsive systems based on line-of-sight tracking, so that they can work efficiently in the real environment.

4. Visual Fatigue and Long-Term Usability

Staring at the system operation for a long time may make people tired of eyes and brains, which will make their performance worse and their experience worse. Unlike hand operation (which will spread the task to different senses), eye control will focus all operations on vision. It may be particularly tiring to keep staring or switching sight targets.

One of the main reasons that make us feel tired is that our eyes should be deliberately stable, which is unnatural. When we look at things, our eyes will unconsciously glance around, but eye-controlled interfaces require us to stare at a small area to trigger the operation. This contradiction between instinctive behavior and mandatory requirements will make eye muscles tense and fatigue faster [3][4].

Physiological research has found some reliable visual fatigue indicators, which will appear when we use eye control equipment. These indicators include more blinking, more frequent changes in pupil size, and changes in saccade frequency-all of which can be monitored in real time [5][9]. For example, Krejtz et al. [5] shows that under the condition of continuous use of eyes, we can well judge whether our brain is tired or not by measuring the changes of pupils.

In order to alleviate these effects, we have come up with several design methods. One way is to reduce staring for a long time and use some instructions that can be completed at a glance, so that you only need to take a quick and natural look to enter. Another common way is to add a rest reminder, a rest timer that can sense our sight, or arrange some gaps that don't require much operation when using for a long time. In addition, some systems can automatically adjust the complexity of the interface when we are tired-for example, make the clickable buttons bigger, simplify the menu layout, or display only the most important tasks.

Another way is to combine eye tracking with other input methods, such as voice, head tracking or gesture control, so that the operation task can be dispersed to different senses and movements. This multi-mode cooperation is particularly useful in an immersive environment like virtual reality (VR), because users are easily tired when they look at 3D images and stare at a place for a long time in this environment.

As machine learning is more and more used in the line-of-sight interactive interface, it becomes more feasible to sense fatigue and adjust the system in real time. Those models trained with physiological and behavioral data can guess how tired users are and then actively change the system behavior. For example, if the user is found to be tired, the system can reduce the interaction frequency, add some short pauses, or adjust the feedback intensity to reduce the burden. Coupled with the federal personalized technology, these adjustments can be fine-tuned according to different people's sensitivity and work endurance.

However, when designing this system, we should pay special attention to everyone's different situation, because everyone's fatigue threshold is very different, and it may be related to factors such as age, vision, light and task type. Therefore, the fatigue perception system that can adapt to changes should be constantly learning and adjusted according to the latest data of users, rather than just relying on fixed thresholds.

In the end, we need a comprehensive approach to make eye-controlled interaction better in long-term use. This includes designing a comfortable interface so that the system can respond intelligently and quickly, and can understand our current state. Future research should also collect data for a long time to see how people's fatigue and eye behavior will change after several days or weeks of continuous use, especially in the scene of auxiliary or barrier-free use.

5. Design Recommendations and Future Directions

According to the challenges discussed above-such as unclear intentions, insufficient feedback, and fatigue accumulation-we will talk about some practical design suggestions in this part, with the aim of making the system controlled by eyes work better. Compared with the previous version, this version has added new content, including intention recognition

based on machine learning, feedback that can be adjusted automatically, and design that can sense personal fatigue, which is consistent with other revised places in the paper.

5.1 Layered Intent Recognition Frameworks:

Rather than relying solely on raw fixation data or rigid dwell-time thresholds, gaze-based systems should implement multi-layered intent recognition models that incorporate contextual cues, fixation dynamics, and task semantics. For example, distinguishing between scanning behavior and deliberate targeting may require combining temporal fixation patterns with spatial clustering and inferred task context. Machine learning models such as LSTM or CNN-based classifiers can be integrated into this framework to improve the accuracy and generalizability of intent recognition across diverse users and scenarios.

5.2 Adaptive Feedback Mechanisms

In order to make us understand what the system is doing, but not feel too complicated, the ways to prompt us should be simple, natural and smart. For example, when we stare at something, the interface can have a small progress bar; Wherever the eyes see, they light up; Or there is a small voice prompt when completing an operation. All these can make us feel more at ease and use it more easily. Also, if the system guesses what we want to do with a ML model that can judge how "sure" we are, then it can adjust the visibility and speed of the prompt according to this "sure" degree, and even decide whether to give the prompt. In this way, the judgment of the system and its response can cooperate well, making the interaction smooth and fast.

5.3 Fatigue-Aware Interaction Design

Fatigue-sensitive design is essential for prolonged or repetitive use of gaze-controlled systems. Systems should minimize the need for extended fixations and instead support glance-based commands and gaze-triggered rest periods. By monitoring real-time signals such as blink frequency, pupil dilation, or microsaccade dynamics, the interface can infer fatigue levels and proactively simplify tasks, dim non-essential content, or prompt rest. With machine learning, especially when combined with historical usage data, systems can further personalize these adaptations based on user-specific fatigue thresholds and behavioral patterns.

5.4 Multimodal Input Synergy

We can't just stare at gaze. Hybrid interactive modes-such as combining eye tracking with head gestures, speech input or touch-can help us understand the user's intention more accurately and reduce false triggers. For example, the system can aim at the target with gaze, and then confirm with voice, which can not only keep hands-free operation, but also improve stability. It is important that these interactions be carefully matched to avoid repeated or conflicting signals. We can also use ML models (machine learning model) to learn the best combination of different interaction modes, which depends on the task type, user behavior and environmental constraints.

5.5 Scenario-Specific Customization

When designing the gaze-based system, we must think about where it will be used. For example, the interface used in medical imaging needs to be very accurate and the response is particularly fast; But the interface used to control smart home is simple and easy to use, and it is not easy to make mistakes. Things like the size of the screen, whether the surrounding light is bright or not, and whether the user will move will all affect how we set the sensitivity of interaction and the speed of feedback. Therefore, there can't be one method that can suit all situations. The gaze interaction platform should bring some adjustable settings or adapt to different situations by itself.

5.6 Federated Learning for Personalization and Privacy

Because everyone's eye habits are different, personalized settings are very important. However, the collection and centralized management of eye data will make people worry about privacy issues. Federated learning technology can train personalized models on our own devices, without sending the original data to the cloud. In this way, everyone can train a shared model together, but everyone's data stays in their own devices. The system can adapt to everyone's habits without revealing our privacy. This method is especially suitable for long-term use in assistive technologies, because such technologies require both privacy protection and precise operation.

6. Conclusion

Gaze-based interaction, as a hands-free natural input method, has great potential in many places, such as assistive technologies, immersive environments, and controlling smart devices. However, there are still some basic problems when it is really used now, such as sometimes it can't figure out what we want to do, the feedback given to us is not clear enough, and the eyes are easily tired after using it for a long time.

This paper systematically evaluates these limitations, and studies the technical solutions and people-centered solutions proposed in the literature. Therefore, it introduces an intention recognition framework based on machine learning, and the simulation experiment proves that compared with the traditional residence time mechanism, the deep neural network model, especially the LSTM classifier, can significantly reduce accidental activation and improve the response ability. Integrating this model can also realize an adaptive feedback system, which can respond intelligently according to prediction confidence, user context and task requirements.

In addition, this paper emphasizes the importance of "fatigue perception design" in long-term use by analyzing the fatigue indexes such as blink frequency and pupil dilation. If these physiological signals are put into the real-time adaptive system, they can make the system adjust the difficulty of interaction in advance, and also actively provide rest opportunities-this can not only reduce fatigue, but also maintain efficiency. The paper also discusses the potential of federated learning, saying that it can establish a personalized line-of-sight model to protect privacy, which may be a development direction in the future.

Generally speaking, these findings show that we need a multi-faceted strategy-this strategy should combine powerful intention modeling, context-aware feedback, multimodal input fusion, and adaptive fatigue management. Such a system should not only be able to identify what users want to do, but also be able to predict when, how and under what conditions interaction should take place.

Looking forward, future gaze interfaces should move beyond static rule-based pipelines toward self-improving, real-time intelligent platforms capable of adapting to individual users and varying environments. This will likely require tighter integration of lightweight machine learning models, edge computing capabilities, and long-term behavioral modeling. Particularly in domains like accessibility and healthcare, the ability to deploy personalized and privacy-conscious gaze systems can fundamentally improve user autonomy, trust, and satisfaction.

With eye tracking technology becoming more and more popular through consumer devices and built-in sensors, the design of the next generation eye-gaze interaction system must balance accuracy, adaptability, transparency and comfort. Only in this way can the interaction mode controlled by eyes really play its full potential in daily use.

REFERENCES

[1] Haans, A., IJsselsteijn, W. A., & de Kort, Y. A. (2008). The virtual midas touch: Helping behavior after a mediated social touch. International Journal of Human-Computer Studies, 66(11), 889–897.

[2] Wedel, M., & Pieters, R. (2008). Eye tracking for visual marketing. Foundations and Trends® in Marketing, 1(4), 231–320.

[3] Duchowski, A. T. (2002). A breadth-first survey of eye-tracking applications. Behavior Research Methods, Instruments, & Computers, 34(4), 455–470.

[4] Duchowski, A. T., & Çöltekin, A. (2007). Foveated gaze-contingent displays for peripheral LOD management, 3D visualization, and stereo imaging. ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM), 3(4), 1–18.

[5] Krejtz, K., Duchowski, A. T., et al. (2018). Eye tracking cognitive load using pupil diameter and microsaccades with fixed gaze. PLOS ONE, 13(9), e0203629.

[6] Kiefer, P., Giannopoulos, I., Raubal, M. (2017). Eye tracking for spatial research: Cognition, computation, challenges. Spatial Cognition & Computation, 17(1–2), 1–19.

[7] Richardson, D. C., & Spivey, M. J. (2004). Eye tracking: Characteristics and methods. In J.M. Henderson & F. Ferreira (Eds.), The interface of language, vision, and action (pp. 17–44). Psychology Press.

[8] Brunyé, T. T., et al. (2019). A review of eye tracking for understanding and improving diagnostic interpretation. Cognitive Research: Principles and Implications, 4(1), 1–16.

[9] Tanenhaus, M. K., & Spivey-Knowlton, M. J. (1996). Eye-tracking. Language and Cognitive Processes, 11(6), 583–588.