electronics

MDPI

Article

Smart Transparency: A User-Centered Approach to Improving Human–Machine Interaction in High-Risk Supervisory Control Tasks

Keran Wang [1], Wenjun Hou [2,3,4,*], Leyi Hong [2] and Jinyu Guo [5]

[1] School of Intelligent Engineering and Automation, Beijing University of Posts and Telecommunications, No. 1 Nanfeng Road, Shahe Higher Education Park, Shahe Area, Changping District, Beijing 102206, China; evakr1206@bupt.edu.cn
[2] School of Digital Media & Design Arts, Beijing University of Posts and Telecommunications, No. 10 Xitucheng Road, Beijing 100876, China; hllly_hyuk@bupt.edu.cn
[3] Beijing Key Laboratory of Network System and Network Culture, No. 10 Xitucheng Road, Beijing 100876, China
[4] Key Laboratory of Interactive Technology and Experience System, Ministry of Culture and Tourism, No. 10 Xitucheng Road, Beijing 100876, China
[5] School of Information and Software Engineering, University of Electronic Science and Technology of China, No. 2006 Xiyuan Avenue, Hi-Tech Zone (West), Chengdu 611731, China; guojinyu@uestc.edu.cn
* Correspondence: hwj1505@bupt.edu.cn

Abstract: In supervisory control tasks, particularly in high-risk fields, operators need to collaborate with automated intelligent agents to manage dynamic, time-sensitive, and uncertain information. Effective human–agent collaboration relies on transparent interface communication to align with the operator's cognition and enhance trust. This paper proposes a human-centered adaptive transparency information design framework (ATDF), which dynamically adjusts the display of transparency information based on the operator's needs and the task type. This ensures that information is accurately conveyed at critical moments, thereby enhancing trust, task performance, and interface usability. Additionally, the paper introduces a novel user research method, Heu–Kano, to explore the prioritization of transparency needs and presents a model based on eye-tracking and machine learning to identify different types of human–agent interactions. This research provides new insights into human-centered explainability in supervisory control tasks.

Keywords: adaptive design; transparency; activity recognition; user research; eye tracking; supervisory control tasks

check for updates

Academic Editors: Jüri Majak, Sónia Sousa and Olga Dunajeva

Received: 16 December 2024
Revised: 14 January 2025
Accepted: 16 January 2025
Published: 21 January 2025

Citation: Wang, K.; Hou, W.; Hong, L.; Guo, J. Smart Transparency: A User-Centered Approach to Improving Human–Machine Interaction in High-Risk Supervisory Control Tasks. *Electronics* **2025**, *14*, 420. https://doi.org/10.3390/electronics14030420

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Supervisory control tasks involve complex situational awareness and high-risk decision-making [1,2], particularly in fields such as Air Traffic Control, where systems demand exceptional reliability and safety. These tasks are characterized by rapidly changing scenarios, large volumes of time-sensitive information, and significant uncertainty. In such dynamic environments, the collaboration between human operators and integrated automated intelligent agents as teammates has become a widely adopted approach [3]. In this collaboration model, human operators typically act as supervisors, processing complex and multidimensional information within a limited time, while intelligent agents assist decision-making by providing predictions or recommendations. However, the final decision ultimately rests with the operator [3,4]. In this context, maintaining cognitive alignment between the operator and the intelligent agent [5], as well as ensuring the operator's

trust in the agent's recommendations [6], is critical for effective human–agent collaboration. To achieve this, the human–agent interface, which serves as a communication and coordination bridge within the human–agent team, must provide transparent information [7]. This transparency helps operators understand the intelligent agent's intentions, capabilities, future plans, and reasoning processes [7,8].

In recent years, increasing research efforts have focused on enhancing the transparency and observability of human–machine collaboration systems. One prominent direction is explainable artificial intelligence (XAI), which emphasizes improving the transparency and interpretability of black-box models, such as deep neural networks [9] and causal inference models [10]. These studies typically adopt the perspective of system developers, focusing on evaluating and interpreting model behavior [11]. Another focus is on human–computer interaction (HCI), where researchers aim to identify and meet users' needs during interactions with intelligent agents [12]. For example, the human–agent transparency model proposed by Lyons [13] and the situation-aware transparency (SAT) model introduced by Chen [14] provide theoretical frameworks for achieving effective transparency in human–machine collaboration [15]. These models also consider operators' cognitive load and information needs in real-world tasks. In terms of interpretation techniques, scholars have explored approaches such as local explanations of task processes [16], feature-based explanations [17], and explanations related to safety, privacy, and vulnerabilities [18,19]. These methods help operators understand the decisions and reasoning processes of AI systems at different stages, enhancing their comprehension of the system and improving decision confidence and collaboration efficiency. Additionally, information visualization and ecological interface design (EID) are widely applied in interface design. These methods use clear, concise, and intuitive text, graphics, and layouts to convey the capabilities, limitations, intentions, and plans of intelligent agents [20–23]. Such designs enable operators to quickly grasp system behavior and improve situational understanding.

Despite significant progress in enhancing system performance, strengthening automated supervision, improving situational awareness, and optimizing trust calibration [5], existing technical explanations often fail to fully account for operators' needs and cognitive processes in real task environments [15]. In complex supervisory control tasks, operators are more concerned with assessing the reliability of system outputs, understanding when to trust the results, and knowing when to remain cautious. These practical needs often take precedence over technical understanding of the model's internal mechanisms [24]. Moreover, operators' demands for transparent information vary according to their cognitive habits, knowledge background, and the context of use. For example, some operators may prefer detailed reasoning processes, while others might focus more on the system's reliability and risk assessments [25]. At the same time, an overabundance of transparency information can lead to information overload, decreasing the efficiency of information retrieval for operators [26]. Research has shown that complex or redundant information displays not only delay decision-making but may also increase the perceived burden of the interface, distract the operator's attention, and even obscure key information [27,28]. Different types of human–machine interactions in supervisory control tasks (e.g., responding to proposals, supervising automation, monitoring automation) [29] require varying levels of transparency information. Therefore, providing transparent information adaptively from a human-centered perspective, based on task scenarios and the operator's personalized needs, has become crucial for improving system usability and trust.

The primary objective of this study is to develop an adaptive transparency information design framework that dynamically adjusts both the presentation and content of transparency information based on operators' personalized needs and task interaction types, with a user-centered perspective. This framework ensures that transparency information is

presented in a timely and appropriate manner when most needed by the operator. Using supervisory control tasks as a case study, the results demonstrate that the framework significantly enhances trust in human–machine collaboration and task performance while maintaining high perceived usability of the interface.

The main contributions of this paper are as follows:

1. Adaptive Transparency Interface Design Framework (ATDF): This study proposes an adaptive transparency interface design framework (ATDF) for situational supervisory control tasks. The framework encompasses two core components: transparency information needs mining and user activity recognition. This framework not only enhances the situational awareness and trust of human–machine teams but also ensures that the perceived usability of the system interface remains at a high level.
2. Heu–Kano User Research Method: A novel user research method, Heu–Kano, is introduced in this study. From a human-centered perspective, it explores the prioritization gradient of transparency information needs in supervisory control tasks.
3. This study presents a non-invasive HAT (human–agent team) interaction-type recognition method based on eye-tracking data and machine learning models. Experimental results demonstrate that this method outperforms existing approaches in terms of task stage recognition accuracy and the required time window length.

The remainder of this paper is organized as follows: Section 2 reviews related work on transparency, user-centered research methods, and non-intrusive task phase recognition. Section 3 provides a detailed description of the Adaptive Transparency Dynamic Framework (ATDF), including the Heu–Kano method and task-type recognition based on eye-tracking and machine learning. Section 4 outlines the experimental design, implementation process, and data collection methods. Section 5 presents the experimental results, including the Heu–Kano computation outcomes and the performance of the machine learning classifiers across datasets with different window lengths. Section 6 discusses the implications of the results and the limitations of the study. Finally, Section 7 concludes the paper by summarizing the key findings and proposing directions for future research.

2. Background

2.1. Human-Centered Explainability and System Transparency Design

Explainable artificial intelligence (XAI) aims to address the transparency issue of AI systems [30]. In recent years, there has been a growing recognition that explainability is not only a technical issue but also a human-centered characteristic, where "explainability depends on the perception and acceptance of the person receiving the explanation" [31]. Although extensive research has improved the understanding of intelligent agent behavior through internal model explanations, these technical explanations often overlook the needs and cognitive processes of users in real task environments [32]. To meet these needs, transparency design should be viewed as a system design issue, not just a technical implementation challenge [33].

Transparency-related information can enhance the predictability and consistency of systems while simultaneously influencing how users perceive, process, and integrate this information into their decision-making, thereby impacting trust formation. However, an overabundance of information may inadvertently increase the cognitive load on operators, negatively affecting the quality of their decisions [26]. This highlights the necessity for transparency design to not only ensure clarity and comprehensibility but also prioritize the delivery of key information that directly supports decision-making processes, thereby fostering a balanced level of reliance and trust.

Establishing a user-centered transparency framework can more effectively assist operators in making timely decisions within dynamic task environments, thereby enhancing human–machine collaboration efficiency. This perspective offers a promising avenue for advancing the field of transparency design.

2.2. Human-Centered User Research Methods (What Transparency Information)

In real-world situational supervisory control tasks, the interaction between humans and intelligent agents is a dynamic process, and the complexity of transparency design arises from the diversity and dynamic nature of user needs. Currently, traditional user research methods, such as design probes [33], interviews [34], the Wizard of Oz method [35], participatory design [36], co-design [37], design work packages [38], and design cards [39], are widely applied to collect information from various dimensions and explore users' perceptions and expectations. However, these methods predominantly rely on qualitative analysis, which makes it difficult to effectively quantify the prioritization of transparency requirements.

Based on situation awareness (SA) theory, Chen et al. [14] proposed the SAT model, which divides transparency information into discrete levels along three dimensions: perception, understanding, and prediction. For example, transparency gradients are represented numerically (such as levels 1-5 in the SAT model). This model aligns with the principles of cognitive psychology and provides scientific guidance for transparency design. However, it primarily defines the grading standards of transparency information at a theoretical level and does not sufficiently account for the diverse needs of users in real-world task environments. Additionally, problem-driven approaches for categorizing requirements (such as the XAI issue library proposed by Liao [31]), while systematically summarizing user needs across various contexts (e.g., "Why did it happen?" and "How can it be optimized?"), fail to address how the gradient of transparency requirements specifically impacts design decisions.

The quantification of transparency requirement prioritization and the adaptive design of diverse needs remain significant challenges, particularly in dynamic task environments with substantial user heterogeneity [25]. Existing research suggests that the Kano model, by quantifying changes in user satisfaction (through the better–worse coefficient), can effectively assess the prioritization of requirements. This model has been widely applied in various fields, such as new energy vehicle user preference analysis [40], digital educational resource demand characterization [41], and video game user modeling [42]. These studies demonstrate the potential of the Kano model to quantify the value of user requirements and provide a theoretical foundation for the prioritization of transparency design. Meanwhile, heuristic evaluation, from the perspective of cognitive psychology, focuses on users' usability feedback and behavioral characteristics [43], offering methodological support for designing transparency information that better aligns with user mental models. These two approaches are typically applied independently in existing studies and have yet to form a systematic methodological framework. By combining the Kano model's quantitative capabilities with the qualitative insights of heuristic evaluation, a novel approach can be established for the excavation and prioritization of transparency requirements in dynamic task scenarios.

2.3. Non-Invasive Task-Type Recognition (What Task Phase to Explain)

Different automation interaction paradigms influence system supervision and intervention. A deep understanding of the applicability of transparent automation to different tasks will provide valuable guidance for engineers developing transparent designs [29]. In dynamic task environments, the effective presentation of transparency information not only requires clarity on "what to explain", but also the precise judgment of "when to explain" [24]. Accurately identifying the user's interaction paradigm (e.g., responding to agent-generated suggestions, supervisory control, or monitoring only) through non-

invasive methods (such as inferring user state from interaction data or behavioral cues), while determining when the user needs transparent information, is key to enhancing the timeliness and user experience of human–machine collaboration systems.

Eye-tracking technology offers a potential solution to this challenge. The classic study by Yarbus [44] demonstrated that eye movement patterns significantly differ across tasks, with eye-tracking data reflecting the user's attention distribution and cognitive state, providing a theoretical basis for inferring task phases. Subsequent studies have further validated this approach. For example, Bulling [45] identified various tasks in office environments using fixation and saccade characteristics, and Wang [46] proposed a self-perception-based gaze task recognition method (TRCLP) for visual task classification in virtual environments. These studies collectively suggest that inferring user task phases from eye-tracking data holds promising application potential.

Eye-tracking data are inherently sequential, and the window length determines the extent of temporal information a model can capture [47]. Shorter time windows are better suited for capturing instantaneous features, such as fixation positions and momentary saccade velocities, while longer windows help models identify long-term dependencies and patterns, such as attention shifts and behavioral transitions. For recurrent neural network-based models like biLSTM [48] and biGRU [49], longer windows are effective at capturing temporal dependencies, though they come at the cost of significantly increased computational and training complexity as the window length grows. In contrast, traditional machine learning models, such as GBDT [50], SVM [51], and RT [52], are generally more computationally efficient. These models do not rely on long-term dependencies and instead use statistical features such as average fixation duration and peak saccade velocity for classification. This makes them particularly efficient in scenarios with smaller datasets or high real-time demands, where they can be quickly trained and effectively deployed.

However, in task scenarios, operators face significant time pressure, and response delays can impact system performance. Therefore, the choice of window length, computational complexity, and real-time requirements must be considered together to strike a balance between timeliness and reliability.

3. Design Framework and Methodology

This section introduces the construction method and details of the Adaptive Transparency Design Framework (ATDF), covering two key components, transparent information needs extraction and user activity recognition, as shown in Figure 1. The ATDF is designed from a human-centered AI perspective, focusing on "ethical design" to explore the gradient of user transparency needs and activity intentions. The framework dynamically adapts the content of transparency information based on these insights.

3.1. Transparent Information Needs Mining Method

To explore the transparent information needs of operators in supervisory control tasks, this study employs a user research method based on heuristic evaluation and the Kano model (Heu–Kano). This method combines usability assessment with user needs analysis, aiming to prioritize the gradient of transparency information requirements while ensuring design usability.

3.1.1. Heuristic Evaluation Stage

Heuristic evaluation involves reviewing the system interface using a standardized usability checklist to identify potential issues in user interactions [43]. In this study, the heuristic evaluation focuses on the presentation of transparency information, aiming to identify, from the perspectives of usability and engineering psychology, which transparency

details are most critical for operators, especially in high-risk and dynamic supervisory control tasks.

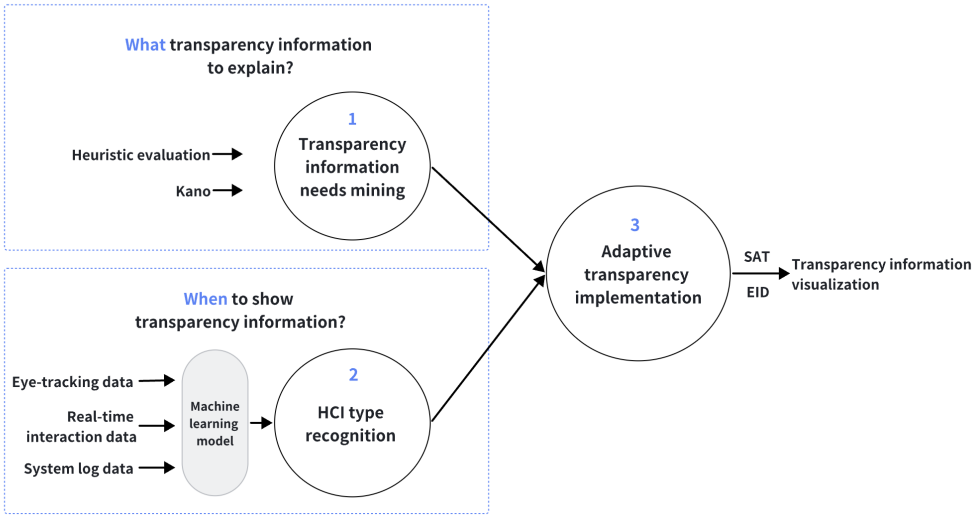


Figure 1. Schematic diagram of the construction and key components within the Adaptive Transparency Design Framework (ATDF).

Based on Nielsen's heuristic evaluation method [53] and considering the characteristics of supervisory control tasks, this study concretizes several guiding indicators (e.g., interpreting "recognition rather than recall" as "the intelligent agent should provide evidence for the user by clarifying its interpretability throughout the decision-making process"). The evaluation covers the operator's needs regarding the system's operational status, task progress, behavioral intentions, risk assessments, and system feedback. A detailed list of these indicators is provided in Appendix A. The evaluation team uses these indicators to assess whether the system interface aligns with usability principles [53]. A list of transparency information pain points is then compiled, and according to Endsey's [54] situational awareness model, the transparency information needs are classified into three dimensions, uncertainty regarding system operation status (perception), concerns over the lack of visibility in task progress (understanding), and confusion due to unclear system intentions (prediction), which will serve as the foundation for subsequent prioritization assessments.

3.1.2. Kano Model Demand Prioritization Classification

The Kano model is a user needs analysis tool used to evaluate how different needs impact user satisfaction [40]. In this study, to ensure data reliability, we designed a survey containing both positive and negative questions. For example, a positive question might be "How would you feel if this feature/function were available?", while a negative question could be "How would you feel if this feature/function were absent?", The full set of questions is provided in Appendix B, covering the three dimensions and 18 items from the pain point list identified in the heuristic evaluation stage.

Each question offers five possible answers: (A) I like it (preferred feature), (B) I expect it (desired feature), (C) I don't care (neutral), (D) I don't like it (disliked feature), (E) I hate it (strongly disliked feature).

After excluding invalid responses and incomplete surveys, we calculated the better and worse coefficients for each demand. This enabled us to quantify the operators' prioritization of transparency information needs, providing the basis for the subsequent adaptive transparency mechanism.

Figure 2 presents the specific implementation steps of the Heu–Kano method.

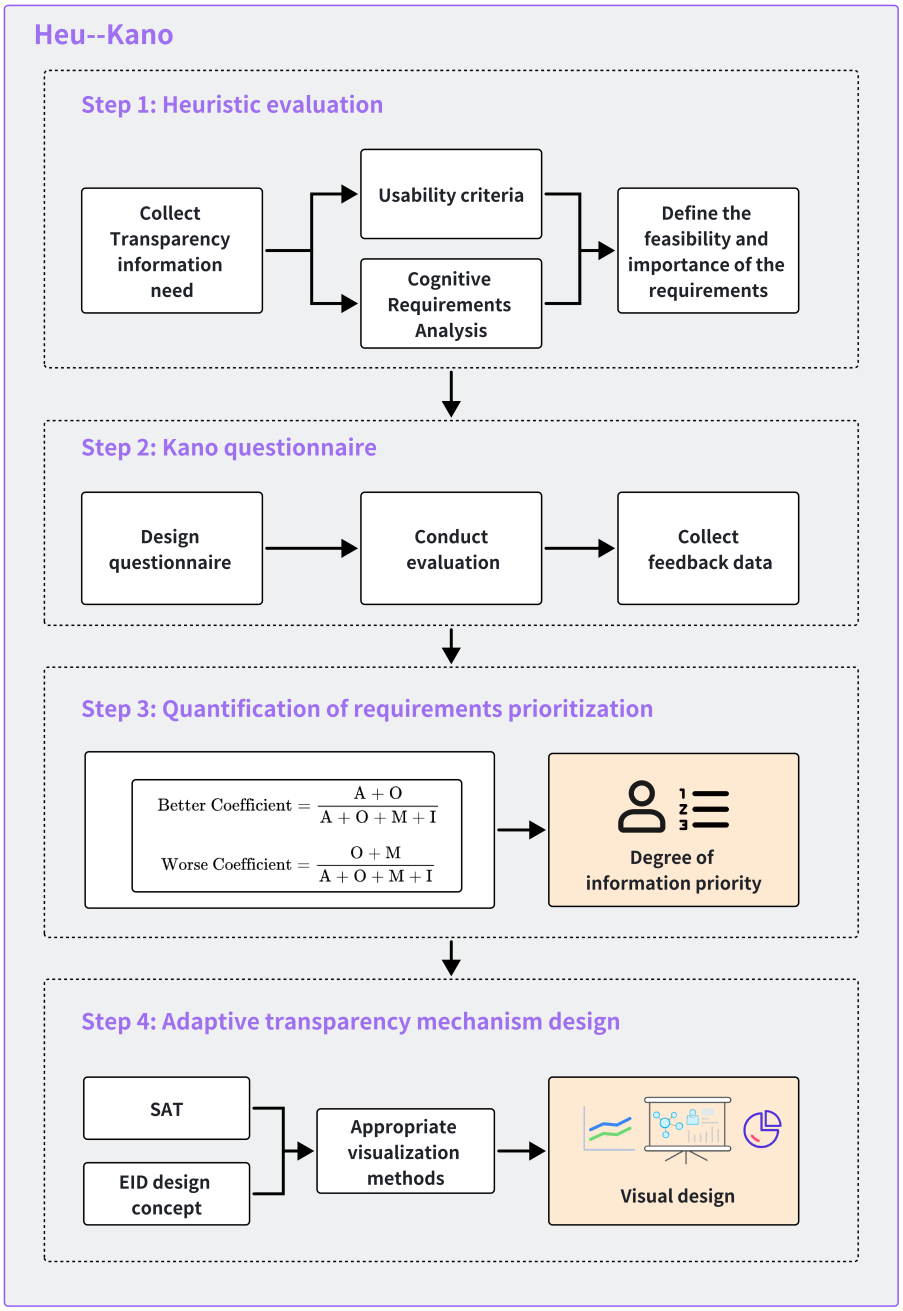


Figure 2. Flowchart of the specific implementation steps of the Heu–Kano method.

3.2. Eye-Tracking-Based Interaction Type Recognition Method

In multi-tasking environments with frequent task switching, activity recognition is crucial for the transparency design of real-time systems. This study proposes an eye-gaze fusion method that combines eye-tracking data, real-time interaction data, and system logs to accurately identify task types. Eye-tracking data reflect the operator's attention distribution, particularly during task switching, where gaze patterns and fixation durations provide key insights into task transitions. By integrating these data with event timestamps from system logs, we can annotate the timing of task activities. To build an efficient task recognition model, we employed traditional machine learning methods such as GBDT, SVM, and decision trees. These methods classify data based on statistical features of eye-tracking, such as fixation duration and saccadic speed, avoiding the need to model temporal dependencies, thus improving computational efficiency. The model's performance

is evaluated using accuracy, precision, recall, and F1 score, with particular attention to the impact of time window length on performance to balance task recognition accuracy with real-time processing requirements.

3.2.1. Time Annotation Data Labeling Method in Multi-Task, Frequent Switching Scenarios

In supervisory control tasks, human–agent interactions primarily include three types: responding to agent-generated recommendations, supervisory control, and passive monitoring [29]. Different interaction paradigms correspond to distinct task allocation strategies, directly influencing the operator's transparency information needs. Additionally, in real-world environments, operators not only interact with agents but also frequently adjust equipment, modify parameters, and communicate with team members. These activities often result in frequent task interruptions, making interaction logging and behavior analysis more complex.

Traditional data annotation methods rely on post hoc annotations, typically requiring experts to manually label data [46]. However, in complex scenarios with frequent task switching, such methods face significant challenges, such as the subjective nature of visual tasks and the difficulty of accurately capturing task-switching moments. These limitations hinder improvements in annotation efficiency and accuracy.

To address these challenges, this study introduces system logs and real-time interaction data as auxiliary tools for data annotation. As illustrated in Figure 3, the annotation process includes the following steps: First, user interaction behaviors, such as mouse click paths and interface scrolling actions, are recorded to preliminarily identify operation intervals. Second, regular expression scripts are employed to extract task-switching patterns and other critical event timestamps from system logs. Finally, the recorded gaze data from eye-tracking devices are integrated to refine temporal boundaries, associating gaze patterns with task types to achieve precise timestamped data annotation.

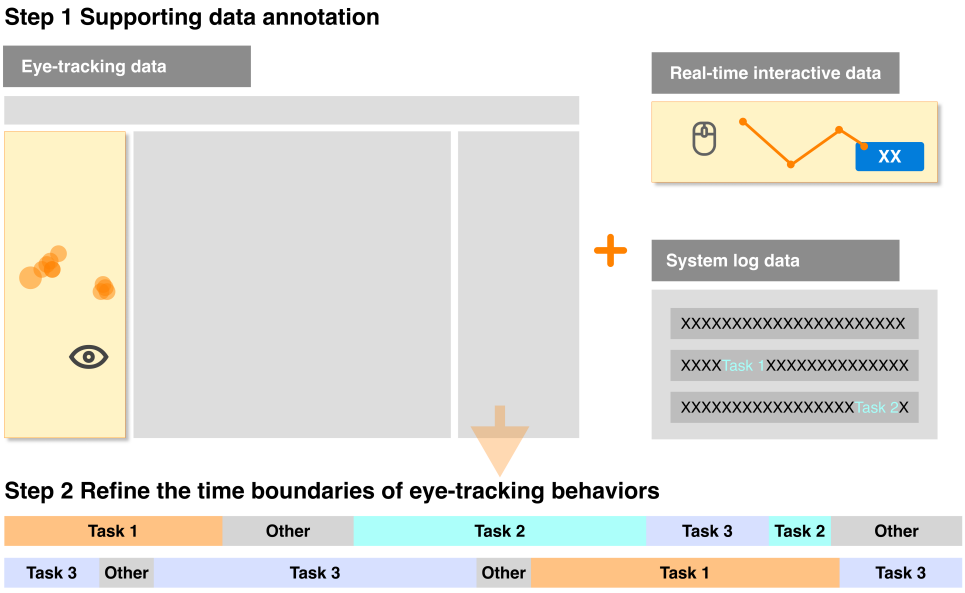


Figure 3. Visualization of system logs and real-time interaction data for labeling in the study. (In Step 1, the yellow circle on the "Eye-tracking data" block represents the operator's gaze point; the yellow line segment on the "real-time interface" block represents the mouse click path; Step 2 shows the duration and execution order of different tasks under experimental processes, and different colored squares represent different tasks).

This annotation method is efficient, easy to implement, and particularly suited for scenarios where log formats are standardized and patterns are well defined. By integrating multi-source data, this approach significantly reduces the complexity of data annotation in scenarios with frequent task switching while ensuring the accuracy and consistency of annotations.

3.2.2. Machine Learning Classifier

As mentioned in the background section, eye-tracking data are inherently sequential, and the window length determines the extent to which the model can capture temporal information. For methods based on recurrent neural networks (such as biLSTM and biGRU), longer windows help capture temporal dependencies but also increase computational costs and training time. In contrast, traditional machine learning models, such as GBDT, SVM, and RT, primarily rely on statistical features like average fixation duration and saccade velocity for classification. By avoiding direct modeling of temporal dependencies, these models offer significant advantages in computational efficiency.

Considering the significant time pressure faced by operators in task scenarios, where response delays may affect system performance, this study prioritizes optimizing the eye-tracking window length and model complexity to strike a balance between real-time performance and reliability while ensuring accurate recognition. Therefore, traditional machine learning classifiers were chosen to improve the system's responsiveness and processing capability, while also meeting both accuracy and real-time requirements.

GBDT (Gradient Boosting Decision Tree) is an ensemble learning method based on an additive model and forward distribution optimization [50]. It improves model performance by progressively building a series of weak learners (typically decision trees) to optimize the loss function.

For multi-class classification problems, the following extended cross-entropy loss function is used:

$$L = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C} y_{i,c}\log(p_{i,c}) \tag{1}$$

where

- C is the total number of classes;
- N represents the total number of samples in the training or testing dataset;
- $y_{i,c}$ is the true label of sample i for class c (using one-hot encoding, it is 1 if the sample belongs to class c; otherwise, it is 0);
- $p_{i,c}$ is the predicted probability from the model that sample i belongs to class c.

The model is initialized with a constant value $F_0(x)$, typically chosen to minimize the log-loss function (e.g., the initial probability is set to the proportion of positive classes in the training set).

In each iteration, GBDT constructs a new decision tree $h_m(x)$ to fit the negative gradient of the current loss function, as follows:

$$g_i^{(m)} = \left.\frac{\partial L}{\partial F(x_i)}\right|_{F=F_{m-1}} \tag{2}$$

where $g_i^{(m)}$ is the negative gradient of the current model for sample i, representing the difference between the model's predicted value and the true label.

The prediction of the newly constructed decision tree $h_m(x)$ is added to the current model as follows:

$$F_m(x) = F_{m-1}(x) + \eta h_m(x) \tag{3}$$

Here, $F_m(x)$ is the updated model after the *m-th* iteration, $F_{m-1}(x)$ is the model from the previous iteration, and η is the learning rate, controlling the contribution of each decision tree to the model. In multi-class tasks, the model's output is transformed into class probabilities through the following softmax function:

$$P(y = c|x) = \frac{\exp(F(x)_c)}{\sum_{k=1}^{C} \exp(F(x)_k)} \tag{4}$$

where

- $P(y = c \mid x)$ is the probability that the sample x belongs to class c;
- $F(x)_c$ is the raw score (logit) of the model for class c;
- C is the total number of classes;
- The denominator is the sum of the exponentials of the raw scores for all classes, ensuring that the output is a valid probability distribution (i.e., all probabilities sum to 1).

In this study, Support Vector Machine (SVM) and decision tree (DT) were selected as baseline methods for performance evaluation in multi-class classification tasks.

SVM achieves classification by finding a hyperplane that maximizes the margin between classes, and it is extended to multi-class tasks using a one-vs.-rest (OvR) strategy [51]. The radial basis function (RBF) kernel was chosen to capture the non-linear features in the data. The model optimization goal is to minimize the trade-off between classification error and margin, with the regularization parameter C and kernel parameter γ tuned using cross-validation.

Decision tree [52] performs classification by recursively splitting the feature space. It uses the Gini Index as the splitting criterion and selects the feature that best discriminates the classes at each node. To control model complexity and avoid overfitting, parameters such as maximum tree depth and minimum samples for a split are used. Multi-class predictions are determined by the class distribution in the leaf nodes of the tree.

3.2.3. Evaluation Metrics

Accuracy, precision, recall, and F1 score are the primary metrics used to evaluate the effectiveness of machine learning models. Higher precision indicates that a model has a better ability to differentiate negative samples, while higher recall reflects that the model has a better ability to distinguish positive samples. The specific calculation methods are shown in the formulas below.

(1) Accuracy

Accuracy is the most intuitive evaluation metric, representing the proportion of correctly predicted samples out of the total samples. Its calculation formula is

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{5}$$

where TP represents true positives (the number of positive samples correctly predicted as positive), TN represents true negatives (the number of negative samples correctly predicted as negative), FP represents false positives (the number of negative samples incorrectly predicted as positive), and FN represents false negatives (the number of positive samples incorrectly predicted as negative).

(2) Precision

Precision measures the accuracy of the model's positive predictions. Specifically, it represents the proportion of samples predicted as positive that are actually positive. Its calculation formula is

$$Precision = \frac{TP}{TP + FP} \tag{6}$$

The higher the precision, the fewer false positives the model produces when predicting positive samples. In practical applications, high precision means that the model's positive predictions are more trustworthy, which is particularly important for tasks requiring highly accurate positive sample identification.

(3) Recall

Recall measures the model's ability to identify positive samples. Specifically, it represents the proportion of actual positive samples that are correctly predicted as positive. Its calculation formula is

$$Recall = \frac{TP}{TP + FN} \tag{7}$$

The higher the recall, the fewer positive samples are missed by the model. Recall is particularly important in tasks such as disease diagnosis or security monitoring, where failing to identify positive samples can lead to serious consequences.

(4) F1 Score

The F1 score is the harmonic mean of precision and recall, providing a single metric that balances both. It is especially useful in situations where there is an imbalance between classes. The F1 score is calculated as

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{8}$$

The F1 score ranges from 0 to 1, with higher values indicating better balance between precision and recall. For tasks that require a balance between precision and recall, the F1 score is a very useful metric.

(5) Required Time Window Length

In eye-tracking data, model performance is not only dependent on classification metrics but also on the chosen time window length. The time window length determines the scope of time information that the model can capture, directly affecting the model's computational efficiency and real-time responsiveness. A shorter time window allows the model to respond more quickly to dynamic operator behaviors, making it suitable for tasks requiring high real-time performance. In contrast, a longer time window can capture longer-term temporal dependencies, improving accuracy but increasing computational complexity.

In this study, we explore the impact of different window lengths on classification performance. By adjusting the window length, the model's accuracy, recall, and F1 score will vary. Both excessively short and long time windows can lead to performance degradation. Therefore, balancing the time window length with model performance is a critical aspect of this research.

4. Experiment

4.1. Participants

In this experiment, participants were recruited through collaborations between the laboratory and industry. After background screening, a total of 50 participants (26 females and 24 males, including 5 usability experts) voluntarily took part in the study. All participants possessed theoretical knowledge of automated supervisory control tasks, with 46 participants having substantial practical experience, and all 5 usability experts being

well versed in heuristic evaluation. Participants included in the data analysis ranged in age from 21 to 40 years (M = 27.1, SD = 2.3). All participants maintained normal or corrected-to-normal vision during the experiment by wearing glasses or contact lenses, with no reported color vision deficiencies.

This study received approval from the Ethics Committee of Beijing University of Posts and Telecommunications prior to data collection. Written informed consent was obtained from all participants, and they were compensated for their participation.

4.2. Experimental Materials

This study designed and developed a simulation system based on an Air Traffic Control (ATC) scenario to emulate complex air traffic monitoring tasks. The system incorporates three typical human–machine interaction tasks along with several auxiliary tasks. Experimental tasks were presented in a complex yet interpretable randomized sequence, with the three primary tasks appearing at approximately equal frequencies. The specific tasks were as follows:

Task 1: Monitoring Only (Aircraft Status Monitoring)

The airspace contained multiple dynamic aircraft (e.g., commercial airplanes, small unmanned aerial vehicles), with their status information (e.g., altitude, speed, heading) displayed in the left or right panels of the system interface. Participants were required to continuously monitor these aircraft, identify whether they were likely to enter designated airspace areas (e.g., "adverse weather zones" or "high traffic density zones"), and promptly detect potential conflicts or anomalies to ensure the stability of airspace operations. This task is illustrated in Figure 4.

Task 2: Responding to Intelligent Recommendations (Scheduling Priority Assessment)

Participants evaluated scheduling priorities by integrating trajectory prediction charts of aircraft with recommendations generated by intelligent agents (e.g., "There is a 90% probability that this aircraft will enter an adverse weather zone; it is recommended to prioritize adjusting its route"). They then adjusted flight paths or parameters as needed to optimize airspace resource allocation and ensure flight safety. This task is illustrated in Figure 5.

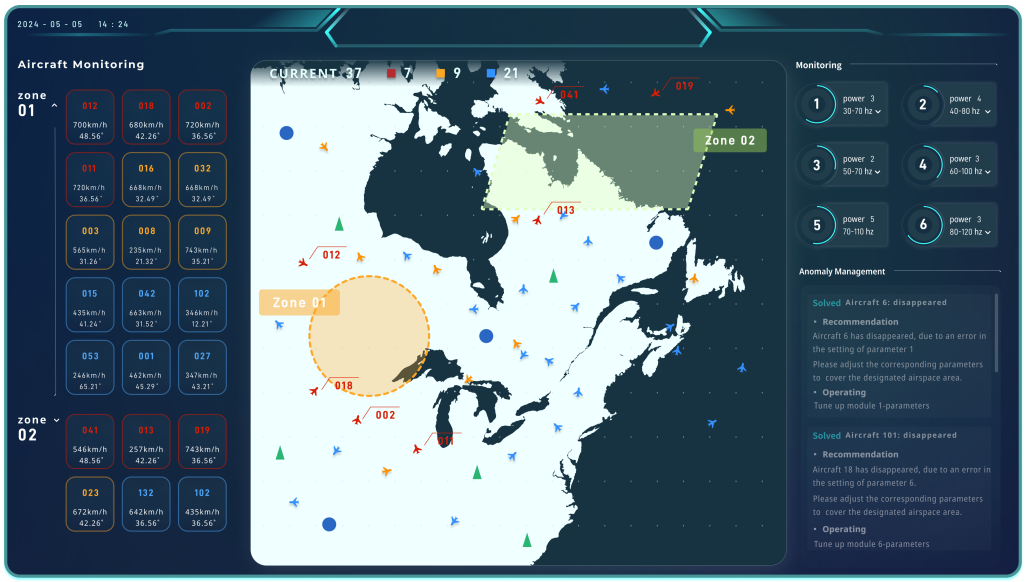


Figure 4. Interface of aircraft status monitoring task experiment. (Different shapes represent various monitoring targets: the airplane symbolizes dynamic aircraft; red, yellow, and blue colors indicate different scheduling priorities; green triangles represent static signal base stations; and blue circles represent static ground radar).

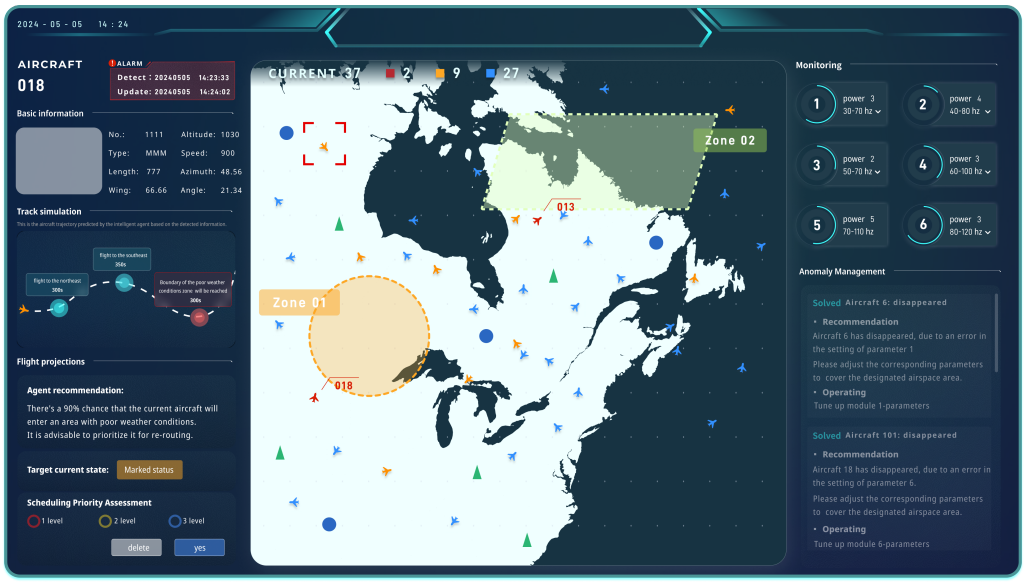


Figure 5. Interface of scheduling priority assessment task experiment. (Different shapes represent various monitoring targets: the airplane symbolizes dynamic aircraft; red, yellow, and blue colors indicate different scheduling priorities; green triangles represent static signal base stations; and blue circles represent static ground radar).

Task 3: Supervising Automation (Parameter Adjustment and Anomaly Management)

When certain aircraft experience abnormal states (e.g., unstable speed, trajectory deviations), the system generates anomaly alerts, including descriptions of the anomalies and the time of occurrence. Participants collaborated with intelligent agents to quickly analyze the causes of these anomalies, adjust flight parameters, or take other necessary actions to restore normal aircraft operations and ensure airspace safety. This task is illustrated in Figure 6.

Additionally, several auxiliary tasks were designed, such as responding to voice prompts and optimizing interface layouts, to enhance the operability and realism of the tasks.

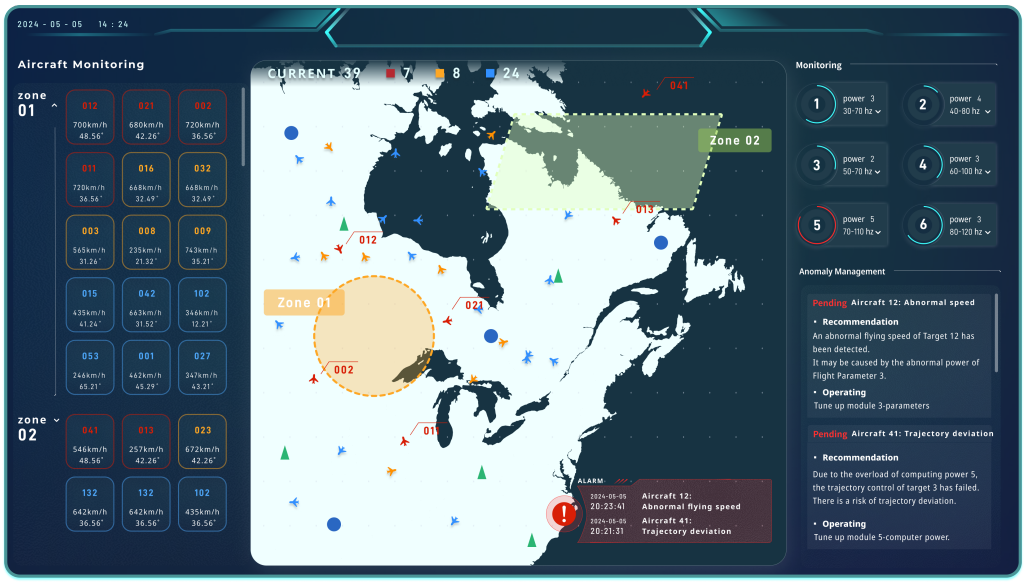


Figure 6. Interface of parameter adjustment and anomaly management task experiment. (Different shapes represent various monitoring targets: the airplane symbolizes dynamic aircraft; red, yellow, and blue colors indicate different scheduling priorities; green triangles represent static signal base stations; and blue circles represent static ground radar).

4.3. Experimental Procedure

Participants assumed the role of Air Traffic Control team members in the experiment, collaborating with intelligent agents within the system to perform real-time monitoring of aircraft, assess scheduling priorities, and manage abnormal states in the airspace. Prior to the start of the experiment, participants underwent comprehensive training on the task objectives and simulation environment to ensure their understanding of the tasks and enhance the consistency of task execution.

Pre-Experiment Practice: Participants underwent a 10-min practice session to familiarize themselves with the simulation system's interface and interaction processes. This practice included a walkthrough of the experimental steps to ensure they understood the objectives and to minimize individual differences that could affect the results.

Formal Experiment: Participants were reminded of the urgency and importance of the tasks and were briefed on the system's reliability. They were instructed to pay close attention to each aircraft that appeared or disappeared and to seriously consider the suggestions provided by the system agent.

Participants completed the experiment in three stages:

Stage 1:

Five usability experts and five general participants were invited to perform tasks using the simulation system and evaluate the transparency of the interface design. Experts observed interaction challenges encountered by the operators and used a modified version of Nielsen's heuristic evaluation checklist [53] to identify issues in the interface design. Meanwhile, all participants provided feedback on the transparency information, indicating which details were helpful for task execution and which were redundant or difficult to interpret. A compiled list of transparency design pain points was created based on these observations (see Appendix A).

Stage 2:

Forty participants completed three rounds of experiments, each including at least three iterations of the three core tasks described earlier. Each experimental round lasted approximately 30 min. After the three rounds, participants completed a Kano questionnaire to evaluate the system's transparency design.

Stage 3:

Twenty participants were invited for an A/B test to compare the adaptive transparency design against a traditional transparency design. This stage assessed which design was more effective based on differences in task performance, cognitive load, and trust. Additional evaluations included the "Human-Automation Trust Scale" [55] and the "NASA TLX Questionnaire" [56] to measure cognitive load and trust. Task performance was evaluated using metrics such as task recognition accuracy and reaction time.

4.4. Dataset

To comprehensively record the interactions of operators at different stages of the tasks, this study employed three data collection methods, eye-tracking, behavior recording, and system logs, as shown in Figure 7. Eye-tracking technology recorded visual metrics such as gaze points and fixation durations, providing insights into operators' attention to transparency information. Behavior recording tracked actions such as mouse clicks and interface scrolling, delineating interaction intervals and clarifying the scope of task activities. Meanwhile, system logs automatically documented timestamps for key task events (e.g., task initiation, pauses, and completion), offering critical data for identifying task transition points.

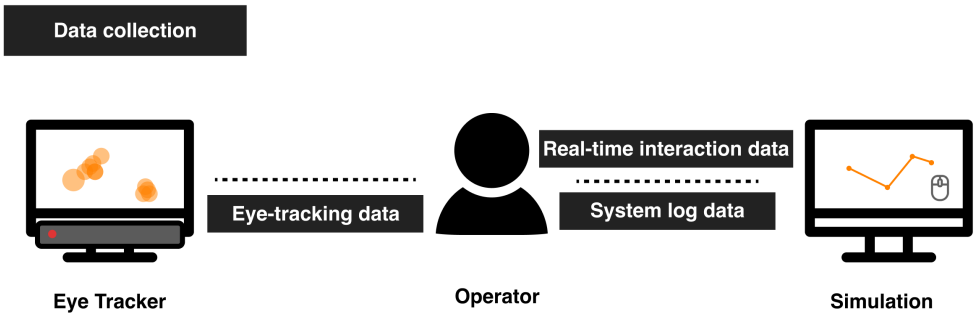


Figure 7. Data collection methods. (The yellow circles in the figure representing the visualization of the operator's gaze points).

The eye-tracking data for this experiment were collected using the Tobii Pro Fusion eye tracker (sampling rate: 250 Hz) and Tobii Pro Lab software (version 1.241) [57], with the screen placed 70 cm from the eye tracker. Prior to each experiment, all participants underwent a standardized calibration procedure to ensure the validity and consistency of the eye-tracking data during the experiment. A laptop (Dell, XPS 8950, Round Rock, TX, USA) was used for data storage and recording. The experiment was conducted on a PC equipped with an Intel Core i7-9700 CPU (Intel Corporation, Santa Clara, CA, USA), an NVIDIA GeForce GTX 3070 Ti GPU (NVIDIA Corporation, Santa Clara, CA, USA) with 8 GB VRAM, and 16 GB DDR4 RAM. The simulation system was based on the Windows 10 operating system [58], with a 19-inch screen, and the experiment was conducted under natural lighting conditions. Additionally, Bandicam software (version 7.1.0) [59] was used for screen recording to capture and save the experiment's process.

After excluding invalid data segments with a sampling rate below 70%, the raw data collected included fixation-related data, saccade-related data, pupil diameter coordinates, timestamps, and task types. Fixation-related data comprised combined fixation origin positions, normalized fixation direction vectors, and corresponding timestamps.

To evaluate the independence of features, we examined their pairwise correlations and calculated Variance Inflation Factors (VIFs). The analysis indicated that the correlations among features were within acceptable thresholds, and all VIF values were below 5, suggesting that multicollinearity or feature dependence would not significantly impact the model's performance. Subsequently, z-score normalization was applied to all features, ensuring a mean of 0 and a standard deviation of 1, thereby reducing the influence of feature scaling on model training.

The eye-tracking device's sampling rate was 250 Hz, and we constructed ten datasets (D1 through D10) with varying time gradients: D1 (0.5 s of eye-tracking data per sample), D2 (1 s), D3 (1.5 s), D4 (2 s), D5 (2.5 s), D6 (3 s), D7 (3.5 s), D8 (4 s), D9 (4.5 s), and D10 (5 s).

In real-world scenarios, operators often face significant time constraints, and system response time directly influences the performance of human–machine teams and user experience. Therefore, beyond the accuracy of the classification model, the length of the time window required by the model must also be considered. The primary objective is to maximize classification performance with a minimal amount of data across multiple tracking experiments while allowing for an exploration of trade-offs. Successfully classifying data from small "fragments" can reduce computational workload and eliminate the need for large-scale experiments.

5. Results

5.1. Heu–Kano-Based Results

Out of the 40 responses received, 38 were deemed valid for analysis after excluding invalid or inconsistent answers. Before conducting the functionality analysis based on the KANO model, SPSS 25.0 [60] was used to test the reliability of the questionnaire. The Cronbach's alpha values for the positive and negative questions were 0.812 and 0.826, respectively, both exceeding the standard threshold of 0.7, indicating good reliability. Additionally, the KMO value for the positive questions was 0.841, with Bartlett's test of sphericity showing significance at $p < 0.001$. For the negative questions, the KMO value was 0.862, also with $p < 0.001$, suggesting that the data were suitable for factor analysis.

According to the classification rules of the Kano model, the category with the highest frequency is considered the classification of the feature. In practical analysis, when the frequencies of two or more categories are close (i.e., the frequency difference is less than 5%), the classification is determined based on the feature's potential impact on user satisfaction, prioritizing categories that provide greater improvements to the user experience. For instance, attractive (A) attributes take precedence over one-dimensional (O) attributes, and one-dimensional attributes are prioritized over indifferent (I) attributes. Additionally, the classification results are further validated by calculating the better–worse coefficients and consulting heuristic evaluations from domain experts to ensure the classification's rationality and consistency.

Figure 8 illustrates the distribution of functionalities in the better–worse matrix. Quadrant A includes attractive attributes such as "highlighting recommended modules" and "adding aircraft trajectory visual information", which were highly valued by users. Their implementation could significantly enhance user experience and elicit positive feedback. Quadrant O contains one-dimensional attributes such as "new aircraft alerts" and "prioritizing high-priority information in overlapping aircraft". These functionalities have a direct impact on user satisfaction, and their optimization would significantly enhance overall satisfaction. Quadrant M includes must-have attributes such as "visualization of operational suggestions" and "detailed presentation of trajectory graphs". These are basic requirements; their absence would lead to dissatisfaction, but fulfilling these needs alone would not increase satisfaction significantly. Quadrant I represents indifferent attributes such as "detailed presentation of map features in trajectory prediction", which have little impact on user satisfaction. Users are generally neutral toward these functionalities, and their improvement or removal would have minimal effect on overall satisfaction.

The experimental results did not identify reverse (R) attributes or questionable (Q) answers, a finding closely tied to the context of this study. In high-risk environments of supervisory control tasks, feature design prioritizes optimizing user experience and supporting decision-making. The functional descriptions in the questionnaire were predominantly framed in a positive manner, making them less likely to elicit negative reactions from users. This task-specific characteristic partially explains the absence of reverse attributes. Additionally, the lack of questionable answers indicates a high level of consistency and rigor in participants' responses to the questionnaire. Future research could explore alternative task scenarios, adjusting the questionnaire's design structure to include assessments of negative attributes, providing a more comprehensive understanding of user needs.

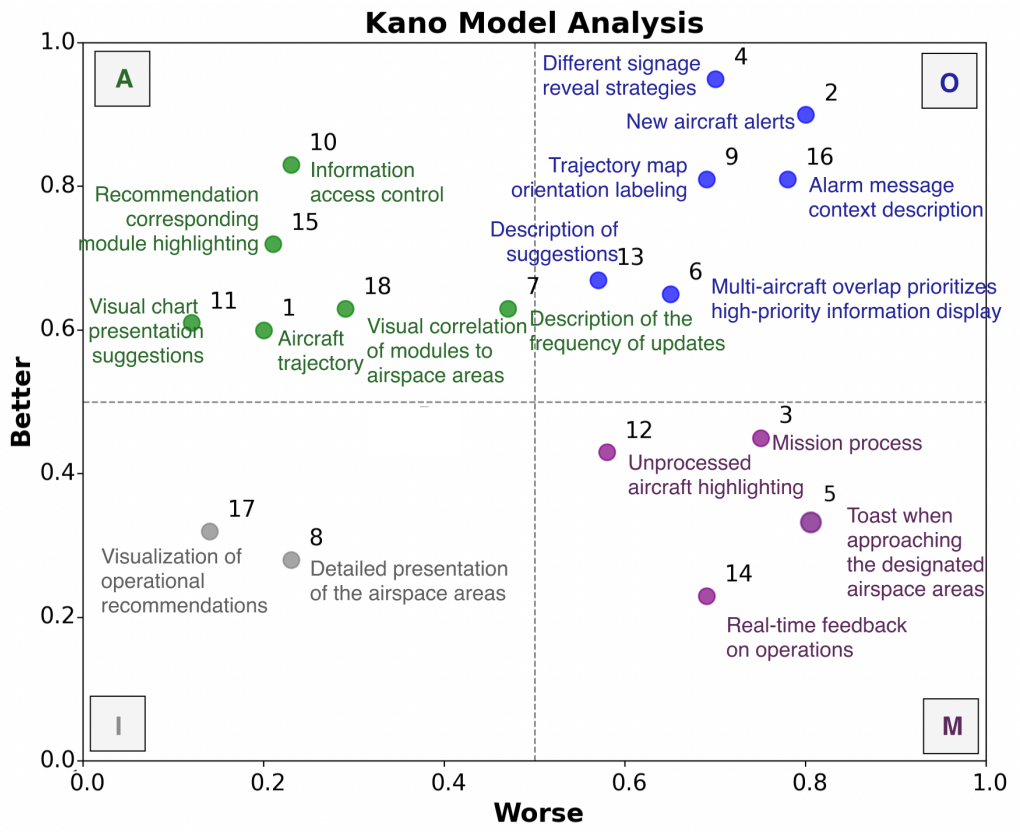


Figure 8. Kano Model Result: Four-Quadrant Distribution Map of better–worse.

5.2. Classifier Performance

Grid search combined with cross-validation was used to select optimal hyperparameter combinations and determine the appropriate ratio between the training and testing datasets.

The number of weak learners (N), the learning rate (ϵ), and the model complexity (C) were the most influential hyperparameters affecting the accuracy of the GBDT model. The value of C indicates the depth of the trees in the GBDT model, with higher C values increasing model complexity. While deeper trees allow the model to capture more intricate patterns, they also increase the risk of overfitting. The learning rate (ϵ) serves as the shrinkage coefficient for each CART (Classification and Regression Tree) in the model and has a range between 0 and 1. A smaller ϵ requires more weak learners to effectively fit the model, potentially improving accuracy but also increasing computational time.

Although the data distribution in this study is relatively balanced, real-world applications often encounter situations where datasets suffer from class imbalance, particularly when instances of certain classes are significantly fewer than those of others. In such cases, traditional evaluation metrics such as accuracy tend to favor the dominant class, failing to adequately reflect the model's ability to recognize under-represented classes. In contrast, the F1 score, which combines precision and recall, provides a more comprehensive assessment of model performance across all classes. It is especially valuable in scenarios with class imbalance, offering a more balanced evaluation of the model's performance. In this study, we evaluated the F1 score performance of GBDT, DT, and SVM models under varying time window lengths. As shown in Figure 9, the performance trends differed among the models as the window length increased.



GBDT	0.331	0.492	0.536	0.774	0.865	0.977	0.978	0.979	0.976	0.971
DT	0.366	0.525	0.647	0.783	0.921	0.931	0.935	0.911	0.876	0.843
SVM	0.211	0.265	0.375	0.469	0.643	0.767	0.879	0.925	0.929	0.934

Figure 9. Model performance (F1 scores) across different window lengths.

For the GBDT model, the F1 score reached its peak at 0.977 with a window length of 6 (D6). Beyond this point, performance gains plateaued, with no significant improvement observed as the window length increased further. The optimal hyperparameter configuration in our experiment was as follows: max depth (max-depth) = 5, learning rate (learning-rate) = 0.01, maximum number of weak learners (n-estimators) = 100, and a training-to-testing dataset ratio of 4:1. These results indicate that the GBDT model effectively captures task features at a window length of 6, and increasing the window length further offers limited performance benefits.

For the DT model, the F1 score was 0.921 at a window length of 5 (D5) and increased to 0.935 at D7, marking the model's best performance. However, as the window length continued to grow, the DT model began to exhibit overfitting, leading to a slight decline in the F1 score. The optimal hyperparameter configuration in our experiment was as follows: learning rate (learning-rate) = 0.05 and maximum number of weak learners (n-estimators) = 100. This suggests that the DT model is sensitive to increasing data volume, with longer windows leading to excessive model complexity tend to overfitting.

For the SVM model, the F1 score reached its maximum of 0.934 at a window length of 10 (D10). Although SVM demonstrated slower convergence, its performance steadily improved with longer time windows. The optimal hyperparameter configuration in our experiment was as follows: kernel type = "linear" penalty coefficient (C) = 1.0, and tolerance for stopping criteria (tol) = 0.001.

5.3. Feature Importance of Eye-Tracking Metrics

Figure 10 presents the feature importance ranking in the classification task, visually reflecting each feature's contribution to model performance through weight values. The results indicate that fixation-related features hold the highest importance in the model, particularly the average fixation duration (fix_dura_avg) and total fixation duration (fix_dura_total), with weights of 0.20 and 0.18, respectively. This highlights that user fixation behavior, especially the length of fixation time, is a decisive factor in task classification. Additionally, the weight of the first fixation duration (first_fix_dura) is

0.15, further emphasizing the significant impact of users' initial fixation behavior on the classification results.

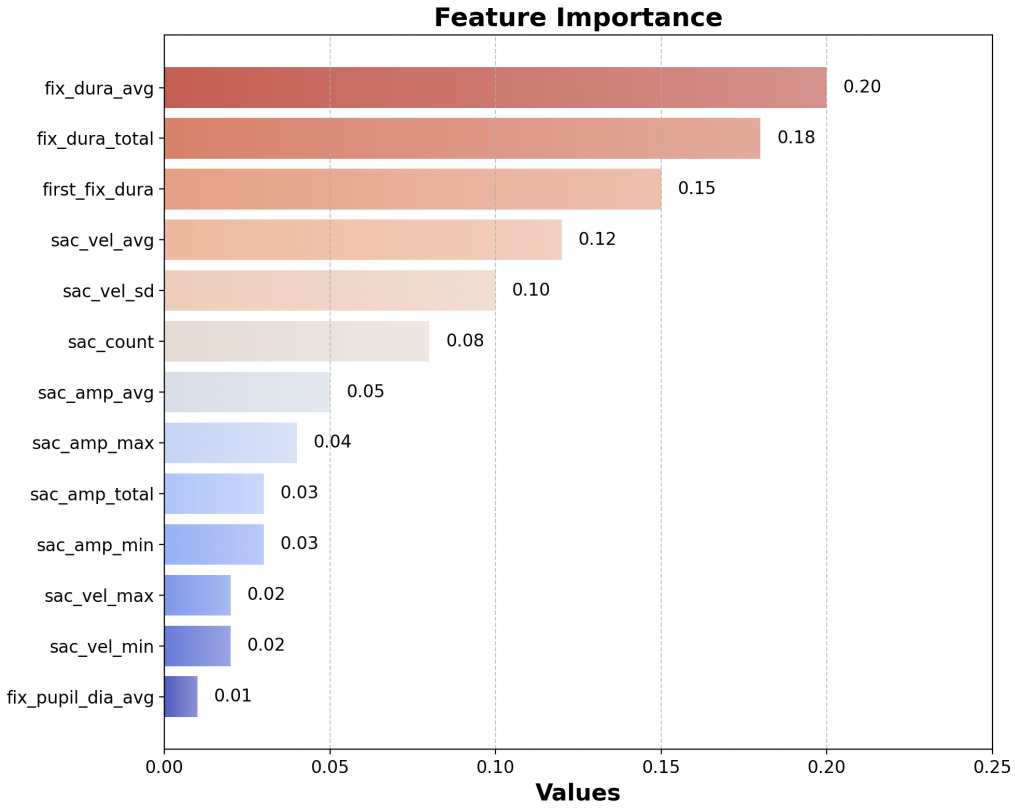


Figure 10. Feature importance in classification tasks.

Saccade-related features follow in importance, ranking below fixation features. Among them, the average saccade velocity (sac_vel_avg) and the standard deviation of saccade velocity (sac_vel_sd) have weights of 0.12 and 0.10, respectively, indicating that saccade velocity is not only a significant feature but that its stability or variability also influences task classification. Meanwhile, the saccade count (sac_count) has a weight of 0.08, demonstrating its auxiliary role in classifying tasks based on users' saccade frequency.

In contrast, saccade amplitude-related features, such as average amplitude (sac_amp_avg) and maximum amplitude (sac_amp_max), exhibit lower importance, with weights of only 0.05 and 0.04. This indicates their limited role in the classification task. Furthermore, the average pupil diameter (fix_pupil_dia_avg) has a minimal weight of 0.01, suggesting that its impact on classification outcomes is relatively weak and that it may not be a critical feature in the current task context.

Overall, fixation features contribute the most to the classification task, especially metrics related to fixation duration, followed by saccade velocity features. In comparison, saccade amplitude and pupil diameter features have relatively lower importance. These findings underscore that fixation behavior and saccade velocity are the most noteworthy feature categories in task classification models, providing clear direction for optimizing future task identification efforts.

5.4. Results of A/B Testing

A/B testing (also known as split testing or controlled experiments) is a commonly used experimental design method for optimization and decision analysis. This study evaluates the differences in task performance, trust, and cognitive load between adaptive interface design and traditional transparency interface design. Task performance was measured

by accuracy and response time, trust was assessed using the "Human-Automation Trust Questionnaire", and cognitive load was evaluated with the "NASA-TLX Cognitive Load Questionnaire" and the "Transparency Information Evaluation Questionnaire".

As shown in Table 1, the results of statistical significance testing indicate that the adaptive design exhibits an advantage in task performance, with an average score of 86.4 (SD = 5.2), which is significantly higher than the 78.3 (SD = 6.1) achieved by the traditional design. An independent samples *t*-test confirmed that the difference was statistically significant ($t = 3.87$, $p < 0.001$). Regarding trust, the adaptive design scored 87.5 (SD = 4.8), higher than the 82.8 (SD = 5.4) of the traditional design, with a significant difference ($t = 2.94$, $p = 0.005$). This suggests that the adaptive design significantly enhances users' trust in the system by providing timely, user-relevant information. For cognitive load, the adaptive design scored 57.3 (SD = 7.2), lower than the 67.9 (SD = 8.1) of the traditional design, and the difference was also significant ($t = -4.12$, $p < 0.001$). This indicates that the adaptive interface effectively reduces cognitive stress during task execution.

To further evaluate the practical significance of the results, Cohen's d was used to calculate effect sizes. The results show medium-to-large effect sizes for task performance ($d = 0.85$), trust ($d = 0.70$), and cognitive load ($d = 0.90$).

In summary, the A/B testing results demonstrate that adaptive interface design significantly improves task performance and user trust while reducing cognitive load by dynamically presenting task-relevant information based on users' interaction behaviors. These findings provide robust empirical support for the value of adaptive interface design in practical applications.

Table 1. Results of A/B testing.

Metric	Adaptive Design (Mean ± SD)	Traditional Design (Mean ± SD)	*t* Value	*p* Value
Task Performance	86.4 ± 5.2	78.3 ± 6.1	3.87	0.001
Confidence Level	87.5 ± 4.8	82.8 ± 5.4	2.94	0.005
Cognitive Load	57.3 ± 7.2	67.9 ± 8.1	−4.12	<0.001

6. Discussion

6.1. Trade-Offs in Time Window Selection

This study employed time windows of varying lengths (ranging from 0.5 to 5 s, in 0.5 s increments) to simulate multi-scale eye-tracking data. The results indicate that a 3 s window (D6) performed best in task classification. This time length balances short-term variability and long-term dependencies within the experimental scope while minimizing noise introduction. Shorter windows (e.g., 0.5 s) can capture fine-grained momentary changes but may miss critical patterns due to insufficient data. Conversely, longer windows (e.g., 5 s) provide rich temporal information but risk diluting task-relevant signals with redundant features and increasing computational complexity.

We hypothesize that this outcome is related to the characteristics of the GBDT model, which incrementally processes non-sequential statistical features. A 3 s window enables the model to effectively capture task-related behavioral variations while avoiding the overfitting risks associated with longer windows in the study's experimental context. While our findings provide empirical support for time window selection, further validation is required across different task scenarios. Additionally, future research could explore real-time dynamic adjustment of time window lengths, such as using shorter windows during highly dynamic task phases to enhance responsiveness and longer windows during stable phases to improve classification accuracy and decision support. This dynamic adjustment

strategy has the potential to increase system adaptability while optimizing computational costs and improving classification performance.

6.2. Classifier Performance and Eye-Tracking Metrics

In selecting classifiers, this study considered factors such as model complexity, computational efficiency, robustness, and dataset size, ultimately evaluating the performance of GBDT, SVM, and DT models. The experimental results demonstrate that under the study's conditions, GBDT excelled in both classification performance and interpretability. Its feature importance ranking validated the potential significance of features such as fixation duration and saccade velocity in task recognition.

Specifically, the average and total fixation durations (fix_dura_avg and fix_dura_total) had the highest feature importance (0.20 and 0.18, respectively), aligning with Yarbus' findings [44], which emphasize the close relationship between attention allocation and task complexity. Additionally, the importance of the first fixation duration (0.15) highlights the role of initial user attention in task classification. Saccade-related features, such as average velocity and velocity standard deviation, ranked second, reflecting users' attention-switching patterns in dynamic tasks. In contrast, features such as saccade amplitude and pupil diameter were less important, potentially reflecting individual differences or environmental factors.

For the SVM model, its performance improved with longer time windows, possibly due to the linear kernel's ability to capture patterns in high-dimensional feature spaces. However, the DT model exhibited overfitting as the time window increased, likely due to high tree complexity and the limited sample size. Future studies could address this by incorporating regularization strategies (e.g., limiting tree depth) and external validation datasets to optimize performance.

6.3. Role of Adaptive Design

Adaptive transparency design dynamically provides relevant information based on users' current task phases and interaction behaviors, effectively reducing cognitive load while enhancing trust in the system. This mechanism improves task efficiency by dynamically optimizing information relevance and timeliness. Our findings align with Endsley's situation awareness theory [14], which underscores the importance of providing appropriate information in high-risk tasks. Furthermore, the high predictive value of fixation duration and first fixation duration indicates that visual attention distribution not only influences users' acceptance of system information but also enhances trust in the interface, thereby improving task performance. Future research could further explore the applicability of adaptive design across diverse task environments, particularly the potential for dynamically adjusting transparency information.

6.4. Limitations and Future Directions

This study has the following limitations that warrant further investigation:

Sample Diversity: The participant sample (n = 50) primarily comprised individuals from academic and industrial backgrounds, which may not fully represent behavioral characteristics across different user groups. This limitation could affect the generalizability of the findings. Future studies should expand sample sizes and include participants from more diverse backgrounds to enhance result applicability.

Risk of Classifier Overfitting: Due to potential feature redundancy, classifiers may exhibit overfitting on small datasets. Future research could optimize model performance through feature selection, regularization, or dimensionality reduction techniques.

Application Scope: The framework was primarily validated within supervisory control tasks. Future studies should explore its application to other domains, such as medical monitoring or financial analysis, to examine its cross-domain applicability.

Dynamic Window Strategy: This study employed fixed time windows. Future research could explore dynamically adjusting time windows based on task state monitoring, which may further optimize classification performance and system efficiency.

7. Conclusions

This study introduces a user-centered adaptive transparency design framework, which integrates the Heu–Kano method with an eye-tracking-based task-type recognition model. The framework dynamically adjusts transparency information based on operator needs and task types, ensuring the accurate delivery of critical information at key moments. Validation within situational supervisory control tasks demonstrates that the framework significantly improves task performance and user trust while effectively reducing cognitive load. These findings provide robust theoretical and practical support for optimizing human–machine collaboration in dynamic task environments.

The Heu–Kano method, as a hybrid tool combining qualitative and quantitative analysis, is systematically applied for the first time to prioritize transparency requirements. This approach offers a scientific basis for dynamic interface design and serves as a methodological reference for other human-centered systems, such as educational resource allocation and medical assistance systems. The method demonstrates innovative potential in addressing diverse user needs.

Furthermore, the study highlights the application potential of eye-tracking technology in task-type recognition. By analyzing critical features such as fixation duration and saccade velocity, the findings reveal a strong correlation between visual behaviors and task phases. This dynamic task recognition capability, grounded in temporal data analysis, supports the real-time adaptation of transparency information and offers valuable technical guidance for other highly dynamic scenarios, including autonomous driving and virtual reality interactions.

Although the study's sample size and application scenarios are limited, the results lay the foundation for further expansion and refinement of the framework. Future research should aim to validate the framework's generalizability by incorporating diverse user samples and task types, exploring dynamic time window strategies, and developing broader task adaptation mechanisms to provide more comprehensive support for transparency design.

Author Contributions: Conceptualization, K.W. and W.H.; Methodology, K.W. and J.G.; Software, K.W., L.H. and J.G.; Validation, L.H.; Formal analysis, K.W., L.H. and J.G.; Investigation, K.W.; Data curation, K.W. and J.G.; Writing—original draft, K.W.; Writing—review & editing, L.H.; Visualization, L.H.; Supervision, K.W.; Project administration, W.H.; Funding acquisition, W.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: This study was conducted in accordance with the Declaration of Helsinki and approved by the Ethics Committee of Beijing University of Posts and Telecommunications, protocol code No. BUPT-P-2024027.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A. Indicator Checklist

This study presents a comprehensive checklist of indicators for supervisory control tasks, comprising a total of 18 items categorized into three dimensions: uncertainty regarding system operation status (perception), concerns related to the lack of visibility in task progress (understanding), and confusion stemming from unclear system intentions (prediction).

Dimensions 1: Uncertainty regarding system operation status (perception).

1. Based on the principle of usability, the airspace areas should include visual information on aircraft trajectories to reduce the monitoring costs for operators.
2. Based on the principles of minimalist design and visibility, the system should adopt different visibility strategies for various categories of goals, prioritizing the information of critical aircraft.
3. Based on the principles of visibility and error prevention, when multiple aircraft exhibit overlapping risks in the airspace areas, the system should highlight critical aircraft and their signboard information to reduce the probability of monitoring errors.
4. Based on the principle of flexibility, the system should provide controls for signboard information visibility and aircraft information presentation according to different user preferences, accommodating various user task execution habits.
5. Based on the principles of usability and proximity, a more relevant design should be adopted for operational suggestions and their corresponding modules, such as highlighting the relevant module when operational suggestions are clicked, thus reducing the cost of information comparison.

Dimensions 2: Concerns related to the lack of visibility in task progress (understanding).

1. Based on the principle of recognition rather than recall, the system should provide effective pop-up feedback prompts for newly introduced aircraft in the airspace areas, rather than relying on the operator's memory.
2. Based on the principle of system state visibility, the system should intuitively display information on monitoring processing progress and task advancement, helping users understand the current phase they are in.
3. Based on the principle of consistency, the image details of the trajectory prediction visualization chart should be exactly the same as the details of the airspace, which would reduce the cost of understanding for the operators.
4. Based on the principle of usability, the trajectory visualization chart should clearly indicate orientation and airspace area information to reduce the identification and comprehension costs for operators.
5. Based on the principle of recognizability, highlighted cues should be added around aircraft whose scheduling priority is undetermined, improving the distinguishability of pending information to prevent user confusion.
6. Based on the principles of visibility and feedback, when operators implement suggestions from agents, the system should provide real-time feedback, helping operators comprehend the effects of each operational step and make gradual adjustments until the task is complete.

Dimensions 3: Confusion stemming from unclear system intentions (prediction).

1. Based on the principle of system state visibility, when critical aircraft approach designated airspace areas, the system should provide a toast notification to alert operators and ensure timely monitoring.
2. Based on the principle of context relevance, the system should offer an intuitive explanation of the frequency of aircraft information updates to help operators understand whether there is a risk of overlooking aircraft.

3. Based on the principles of minimalist design and visibility, suggested information from agents should be more prominently displayed, potentially using visual charts to convey information intuitively.
4. Based on the principle of matching user expectations, necessary explanations regarding parameter adjustment and anomaly management suggestions' transparency should be provided to aid user understanding.
5. Based on the principle of context relevance, anomaly management information should provide clearer contextual explanations to assist operators in intuitively understanding system failures and their causes.
6. Based on the principles of information distinguishability and comprehensibility, the textual information related to current agent suggestions, which may be excessive, should be moderately refined or enhanced through visual means to improve information discernibility and facilitate quick understanding of agent suggestions.
7. Based on the principle of context relevance, the system should supplement each control module with necessary functional explanations that relate to the airspace areas, assisting operators in understanding the module functions and their operational scope.

Appendix B. Kano Questionnaire Description

In this study, a 36-question Kano test was conducted to evaluate 18 functional points. A sample of the questionnaire items designed and utilized is presented in Table A1:

Table A1. A sample of the Functional Points Questionnaire.

Functional Points: Add visual information of aircraft trajectories within the situational picture.					
	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
How would you feel if this feature was available?					
How would you feel if this feature was not available?					

The specific functional points include the following 18 items:

1. Add visual information of aircraft trajectories within the airspace areas.
2. Provide an alert bar on the side of the system for new aircraft in the status map.
3. Display monitoring processing progress and task progress information at the top of the status graph.
4. Adopt different labeling strategies for various categories of aircraft, prioritizing the highlighting of aircraft information.
5. Provide toast alerts when crucial aircraft are approaching the designated airspace areas.
6. Prioritize the presentation of crucial aircraft and their signage information when there are overlapping risks among multiple aircraft in the airspace areas.
7. The system provides an intuitive explanation of the update frequency of aircraft information.
8. The visual chart of track predictions should illustrate local details of the airspace area map, such as terrain and rivers.
9. The track prediction visualization chart should be clearly marked with bearing and designated airspace area information.
10. Provide controls for the display and concealment of signage information and the presentation of aircraft information according to different user preferences.
11. Use visual charts instead of text to present agent suggestions.
12. Add highlighting alerts around aircraft whose scheduling priority is undetermined.
13. Provide detailed descriptions of parameter adjustment and anomaly management suggestions.
14. Offer real-time status feedback when the operator executes an agent suggestion.

15. Highlight the corresponding operational module when an operation suggestion is clicked.
16. Provide clear contextual information for anomaly management suggestions.
17. Present operational recommendations visually.
18. Create a visual association of control modules with the airspace areas.

References

1. Inga, J.; Ruess, M.; Robens, J.H.; Nelius, T.; Rothfuß, S.; Kille, S.; Dahlinger, P.; Lindenmann, A.; Thomaschke, R.; Neumann, G.; et al. Human-machine symbiosis: A multivariate perspective for physically coupled human-machine systems. *Int. J. Hum.-Comput. Stud.* **2023**, *170*, 102926. [CrossRef]
2. Bach, T.A.; Khan, A.; Hallock, H.; Beltrão, G.; Sousa, S. A systematic literature review of user trust in AI-enabled systems: An HCI perspective. *Int. J. Hum.-Comput. Interact.* **2024**, *40*, 1251–1266. [CrossRef]
3. Chignell, M.; Wang, L.; Zare, A.; Li, J. The evolution of HCI and human factors: Integrating human and artificial intelligence. *Acm Trans. Comput.-Hum. Interact.* **2023**, *30*, 1–30. [CrossRef]
4. Steyvers, M.; Kumar, A. Three challenges for AI-assisted decision-making. *Perspect. Psychol. Sci.* **2024**, *19*, 722–734. [CrossRef]
5. Endsley, M.R. Supporting Human-AI Teams: Transparency, explainability, and situation awareness. *Comput. Hum. Behav.* **2023**, *140*, 107574. [CrossRef]
6. Mehrotra, S.; Degachi, C.; Vereschak, O.; Jonker, C.M.; Tielman, M.L. A systematic review on fostering appropriate trust in human-AI interaction. *arXiv* **2023**, arXiv:2311.06305.
7. Eyert, F.; Lopez, P. Rethinking transparency as a communicative constellation. In Proceedings of the 2023 ACM Conference on Fairness, Accountability, and Transparency, Chicago, IL, USA, 12–15 June 2023; pp. 444–454.
8. Bhaskara, A.; Skinner, M.; Loft, S. Agent transparency: A review of current theory and evidence. *IEEE Trans. Hum.-Mach. Syst.* **2020**, *50*, 215–224. [CrossRef]
9. Montavon, G.; Samek, W.; Müller, K.R. Methods for interpreting and understanding deep neural networks. *Digit. Signal Process.* **2018**, *73*, 1–15. [CrossRef]
10. Joshi, G.; Walambe, R.; Kotecha, K. A review on explainability in multimodal deep neural nets. *IEEE Access* **2021**, *9*, 59800–59821. [CrossRef]
11. Kostopoulos, G.; Davrazos, G.; Kotsiantis, S. Explainable Artificial Intelligence-Based Decision Support Systems: A Recent Review. *Electronics* **2024**, *13*, 2842. [CrossRef]
12. Rong, Y.; Leemann, T.; Nguyen, T.T.; Fiedler, L.; Qian, P.; Unhelkar, V.; Seidel, T.; Kasneci, G.; Kasneci, E. Towards human-centered explainable ai: A survey of user studies for model explanations. *IEEE Trans. Pattern Anal. Mach. Intell.* **2023**, *46*, 2104–2122. [CrossRef] [PubMed]
13. Lyons, J.B. Being transparent about transparency: A model for human-robot interaction. In Proceedings of the 2013 AAAI Spring Symposium Series, Palo Alto, CA, USA, 25–27 March 2013.
14. Chen, J.Y.; Procci, K.; Boyce, M.; Wright, J.; Garcia, A.; Barnes, M. Situation awareness-based agent transparency. *Army Res. Lab.* **2014**, 1–29. Available online: https://apps.dtic.mil/sti/citations/ADA600351 (accessed on 15 December 2024).
15. Liao, Q.V.; Varshney, K.R. Human-centered explainable ai (xai): From algorithms to user experiences. *arXiv* **2021**, arXiv:2110.10790.
16. Xie, Y.; Pongsakornsathien, N.; Gardi, A.; Sabatini, R. Explanation of machine-learning solutions in air-traffic management. *Aerospace* **2021**, *8*, 224. [CrossRef]
17. Antoniadi, A.M.; Du, Y.; Guendouz, Y.; Wei, L.; Mazo, C.; Becker, B.A.; Mooney, C. Current challenges and future opportunities for XAI in machine learning-based clinical decision support systems: A systematic review. *Appl. Sci.* **2021**, *11*, 5088. [CrossRef]
18. Sokol, K.; Flach, P. Explainability fact sheets: A framework for systematic assessment of explainable approaches. In Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency, Barcelona, Spain, 27–30 January 2020; pp. 56–67.
19. Rayo, M.F.; Horwood, C.R.; Fitzgerald, M.C.; Grayson, M.R.; Abdel-Rasoul, M.; Moffatt-Bruce, S.D. Situated visual alarm displays support machine fitness assessment for nonexplainable automation. *IEEE Trans. Hum.-Mach. Syst.* **2022**, *52*, 984–993. [CrossRef]
20. Dodeja, L.; Tambwekar, P.; Hedlund-Botti, E.; Gombolay, M. Towards the design of user-centric strategy recommendation systems for collaborative Human–AI tasks. *Int. J. Hum.-Comput. Stud.* **2024**, *184*, 103216. [CrossRef]
21. Akhtar, M.A.K.; Kumar, M.; Nayyar, A. The Role of Human-Centered Design in Developing Explainable AI. In *Towards Ethical and Socially Responsible Explainable AI: Challenges and Opportunities*; Springer: Berlin/Heidelberg, Germany, 2024; pp. 99–126.
22. Selkowitz, A.R.; Larios, C.A.; Lakhmani, S.G.; Chen, J.Y. Displaying information to support transparency for autonomous platforms. In *Advances in Human Factors in Robots and Unmanned Systems: Proceedings of the AHFE 2016 International Conference on Human Factors in Robots and Unmanned Systems, Walt Disney World®, FL, USA, 27–31 July 2016*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 161–173.

23. Kilgore, R.; Voshell, M. Increasing the transparency of unmanned systems: Applications of ecological interface design. In *Virtual, Augmented and Mixed Reality. Applications of Virtual and Augmented Reality: Proceedings of the 6th International Conference, VAMR 2014, Held as Part of HCI International 2014, Heraklion, Crete, Greece, 22–27 June 2014*; Proceedings, Part II 6; Springer: Berlin/Heidelberg, Germany, 2014; pp. 378–389.
24. Ma, S. Towards Human-centered Design of Explainable Artificial Intelligence (XAI): A Survey of Empirical Studies. *arXiv* **2024**, arXiv:2410.21183.
25. Nimmo, R.; Constantinides, M.; Zhou, K.; Quercia, D.; Stumpf, S. User Characteristics in Explainable AI: The Rabbit Hole of Personalization? In Proceedings of the CHI Conference on Human Factors in Computing Systems, Honolulu, HI, USA, 11–16 May 2024; pp. 1–13.
26. Phillips-Wren, G.; Adya, M. Decision making under stress: The role of information overload, time pressure, complexity, and uncertainty. *J. Decis. Syst.* **2020**, *29*, 213–225. [CrossRef]
27. Mai, Q. Transparency is a Misplaced Regulatory Focus for Holding Adaptive Software as Medical Devices (SaMDs) Accountable. *Boston Coll. Intellect. Prop. Technol. Forum* **2023**, *2023*, 1–21.
28. Zhang, Y.; Wang, W.; Zhou, X.; Wang, Q.; Sun, X. Tactical-level explanation is not enough: Effect of explaining AV's lane-changing decisions on drivers' decision-making, trust, and emotional experience. *Int. J. Hum.-Comput. Interact.* **2023**, *39*, 1438–1454. [CrossRef]
29. van de Merwe, K.; Mallam, S.; Nazir, S. Agent transparency, situation awareness, mental workload, and operator performance: A systematic literature review. *Hum. Factors* **2024**, *66*, 180–208. [CrossRef] [PubMed]
30. Kim, J.; Maathuis, H.; Sent, D. Human-centered evaluation of explainable AI applications: A systematic review. *Front. Artif. Intell.* **2024**, *7*, 1456486. [CrossRef] [PubMed]
31. Liao, Q.V.; Gruen, D.; Miller, S. Questioning the AI: Informing design practices for explainable AI user experiences. In Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems, Honolulu, HI, USA, 25–30 April 2020; pp. 1–15.
32. Bertrand, A.; Belloum, R.; Eagan, J.R.; Maxwell, W. How cognitive biases affect XAI-assisted decision-making: A systematic review. In Proceedings of the 2022 AAAI/ACM Conference on AI, Ethics, and Society, San Jose, CA, USA, 21–23 October 2022; pp. 78–91.
33. Yu, S. Towards Trustworthy and Understandable AI: Unraveling Explainability Strategies on Simplifying Algorithms, Appropriate Information Disclosure, and High-level Collaboration. In Proceedings of the 26th International Academic Mindtrek Conference, Tampere, Finland, 3–6 October 2023; pp. 133–143.
34. Wang, Q.; Camacho, I.; Jing, S.; Goel, A.K. Understanding the Design Space of AI-Mediated Social Interaction in Online Learning: Challenges and Opportunities. *ACM Hum.-Comput. Interact.* **2022**, *6*, 1–26. [CrossRef]
35. Xin, D.; Wu, E.Y.; Lee, D.J.L.; Salehi, N.; Parameswaran, A. Whither automl? understanding the role of automation in machine learning workflows. In Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems, Virtual, 8–13 May 2021; pp. 1–16.
36. Viswanathan, S.; Boulard, C.; Bruyat, A.; Maria Grasso, A. Situational Recommender: Are You On the Spot, Refining Plans, or Just Bored? In Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems, New Orleans, LA, USA, 29 April–5 May 2022; pp. 1–19.
37. Lee, M.K.; Kusbit, D.; Kahng, A.; Kim, J.T.; Yuan, X.; Chan, A.; See, D.; Noothigattu, R.; Lee, S.; Psomas, A.; et al. WeBuildAI: Participatory framework for algorithmic governance. *ACM Hum.-Comput. Interact.* **2019**, *3*, 1–35. [CrossRef]
38. Biermann, O.C.; Ma, N.F.; Yoon, D. From tool to companion: Storywriters want AI writers to respect their personal values and writing strategies. In Proceedings of the 2022 ACM Designing Interactive Systems Conference, Virtual Event, 13–17 June 2022; pp. 1209–1227.
39. Lee, C.P.; Lee, M.K.; Mutlu, B. The AI-DEC: A Card-based Design Method for User-centered AI Explanations. In Proceedings of the 2024 ACM Designing Interactive Systems Conference, Copenhagen, Denmark, 1–5 July 2024; pp. 1010–1028.
40. Yang, Y.; Li, Q.; Li, C.; Qin, Q. User requirements analysis of new energy vehicles based on improved Kano model. *Energy* **2024**, *309*, 133134. [CrossRef]
41. Feng, R.; Luo, J. Analysis of K-12 teachers' demand characteristics for digital educational resource services from the perspective of adaptive feedback: Based on the analysis of KANO model. *Educ. Inf. Technol.* 2024, *in press*. [CrossRef]
42. Liu, Y.; You, T.H.; Zou, J.; Cao, B.B. Modelling customer requirement for mobile games based on online reviews using BW-CNN and S-Kano models. *Expert Syst. Appl.* **2024**, *258*, 125142. [CrossRef]
43. Fernández, J.; Macías, J.A. Heuristic-based usability evaluation support: A systematic literature review and comparative study. In Proceedings of the XXI International Conference on Human Computer Interaction, Málaga, Spain, 22–24 September 2021; pp. 1–9.
44. Yarbus, A.L. *Eye Movements and Vision*; Springer: Berlin/Heidelberg, Germany, 2013.
45. Bulling, A.; Ward, J.A.; Gellersen, H.; Tröster, G. Eye movement analysis for activity recognition using electrooculography. *IEEE Trans. Pattern Anal. Mach. Intell.* **2010**, *33*, 741–753. [CrossRef]

46. Wang, Z.; Lu, F. Tasks Reflected in the Eyes: Egocentric Gaze-Aware Visual Task Type Recognition in Virtual Reality. *IEEE Trans. Vis. Comput. Graph.* **2024**, *30*, 7277–7287. [CrossRef]
47. Arshad, M.H.; Bilal, M.; Gani, A. Human activity recognition: Review, taxonomy and open challenges. *Sensors* **2022**, *22*, 6463. [CrossRef] [PubMed]
48. Elmadjian, C.; Gonzales, C.; Costa, R.L.d.; Morimoto, C.H. Online eye-movement classification with temporal convolutional networks. *Behav. Res. Methods* **2023**, *55*, 3602–3620. [CrossRef] [PubMed]
49. Niu, M.; Zheng, Z.K.; Akash, K.; Misu, T. Beyond empirical windowing: An attention-based approach for trust prediction in autonomous vehicles. In Proceedings of the ICASSP 2024–2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Seoul, Republic of Korea, 14–19 April 2024; IEEE: New York, NY, USA, 2024; pp. 5615–5619.
50. Wang, S.; Ma, J. A novel GBDT-BiLSTM hybrid model on improving day-ahead photovoltaic prediction. *Sci. Rep.* **2023**, *13*, 15113. [CrossRef] [PubMed]
51. Baskaran, P.; Smith, J.B.; Adams, J.A. Visual task recognition for human-robot teams. In Proceedings of the 2022 IEEE 3rd International Conference on Human-Machine Systems (ICHMS), Orlando, FL, USA, 17–19 November 2022; IEEE: New York, NY, USA, 2022; pp. 1–6.
52. Wang, K.; Hou, W.; Ma, H.; Hong, L. Eye-Tracking Characteristics: Unveiling Trust Calibration States in Automated Supervisory Control Tasks. *Sensors* **2024**, *24*, 7946. [CrossRef]
53. Nielsen, J.; Molich, R. Heuristic evaluation of user interfaces. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Seattle, WA, USA, 1–5 April 1990; pp. 249–256.
54. Endsley, M.R. Situation awareness. In *Handbook of Human Factors and Ergonomics*; John Wiley & Sons, Inc.: Hoboken, NJ, USA, 2021; pp. 434–455.
55. Gulati, S.; Sousa, S.; Lamas, D. Design, development and evaluation of a human-computer trust scale. *Behav. Inf. Technol.* **2019**, *38*, 1004–1015. [CrossRef]
56. Hart, S.G.; Staveland, L.E. Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research. *Adv. Psychol.* **1988**, *52*, 139–183. [CrossRef]
57. Tobii Technology. *Tobii Pro Fusion User Manual*; Technical Report; Tobii Technology: Stockholm, Sweden, 2023. Available online: https://go.tobii.com/tobii-pro-fusion-user-manual (accessed on 21 July 2024) .
58. Microsoft Corporation. *Windows 10: A Guide to the New Features*; Microsoft Corporation: Redmond, WA, USA, 2024. Available online: https://www.microsoft.com/windows (accessed on 18 July 2024).
59. Bandicam Company. Bandicam Company. Bandicam Screen Recorder. 2008. Available online: https://www.bandicam.com (accessed on 21 July 2024) .
60. IBM Corporation. *IBM SPSS Statistics 28.0 User's Guide*; Technical Report; IBM Corporation: Armonk, NY, USA, 2021. Available online: https://www.ibm.com/docs/en/SSLVMB_28.0.0/pdf/IBM_SPSS_Statistics_Core_System_User_Guide.pdf (accessed on 21 September 2024) .

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.