

Article

Eye-Tracking Characteristics: Unveiling Trust Calibration States in Automated Supervisory Control Tasks

Keran Wang ¹, Wenjun Hou ^{2,3,4,*}, Huiwen Ma ² and Leyi Hong ²

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@163.com

² School of Digital Media & Design Arts, Beijing University of Posts and Telecommunications, No. 10 Xitucheng Road, Beijing 100876, China

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

* Correspondence: hwj1505@bupt.edu.cn

Abstract: Trust is a crucial human factor in automated supervisory control tasks. To attain appropriate reliance, the operator's trust should be calibrated to reflect the system's capabilities. This study utilized eye-tracking technology to explore novel approaches, given the intrusive, subjective, and sporadic characteristics of existing trust measurement methods. A real-world scenario of alarm state discrimination was simulated and used to collect eye-tracking data, real-time interaction data, system log data, and subjective trust scale values. In the data processing phase, a dynamic prediction model was hypothesized and verified to deduce and complete the absent scale data in the time series. Ultimately, through eye tracking, a discriminative regression model for trust calibration was developed using a two-layer Random Forest approach, showing effective performance. The findings indicate that this method may evaluate the trust calibration state of operators in human–agent collaborative teams within real-world settings, offering a novel approach to measuring trust calibration. Eye-tracking features, including saccade duration, fixation duration, and the saccade–fixation ratio, significantly impact the assessment of trust calibration status.



Citation: Wang, K.; Hou, W.; Ma, H.; Hong, L. Eye-Tracking Characteristics: Unveiling Trust Calibration States in Automated Supervisory Control Tasks. *Sensors* **2024**, *24*, 7946. <https://doi.org/10.3390/s24247946>

Academic Editor: Klaas Bombeke

Received: 28 October 2024

Revised: 27 November 2024

Accepted: 10 December 2024

Published: 12 December 2024



Copyright: © 2024 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/>).

Keywords: automated supervisory control; trust measurement; trust calibration; eye tracking; Random Forest

1. Introduction

Automated supervisory control tasks often involve high-risk decision making, particularly in the aeronautical-military domain [1,2], characterized by quickly changing mission scenarios, time-sensitive information, and significant uncertainty and openness. The human operator typically performs as a supervisor [3], rapidly processing complex, multidimensional information and collaborating with automation-integrated intelligent agents to enhance decision-making outcomes in joint surveillance scenarios. Trust, defined as “the attitude that an agent will help achieve an individual's goals in a situation characterized by uncertainty and vulnerability”, is essential in shaping human–agent interaction in this context [4]. On one hand, a lack of trust compels operators to invest additional effort in verifying the accuracy of information provided by agents [5], potentially leading to decreased decision-making performance in team tasks and even the neglect or misjudgment of critical information [6]. On the other hand, excessive trust may reduce operators' vigilance in active monitoring [7], thereby increasing unknown risks and potentially causing safety incidents. Therefore, real-time assessment of operators' trust calibration status—ensuring that perceived trust aligns with the actual capabilities and reliability of the agent [8]—and

implementing strategic adjustments when operators exhibit over-trust or under-trust are crucial. These measures are essential for fostering higher-quality interactions between operators and intelligent agents, improving performance in situational supervisory control tasks, and mitigating human–agent conflicts arising from trust biases [9,10].

Accurately measuring trust in automated systems is essential for trust calibration. Researchers have proposed various methods for assessing trust. (1) Self-report scales: These are the most commonly used tools for measuring user trust due to their intuitive nature, cost-effectiveness, and ease of use [11,12]. Self-report scales reliably capture users' perceived trust at a specific point in time, often at the conclusion of an experiment. However, they do not provide continuous measurements and are therefore unable to capture real-time trust fluctuations. Additionally, the data collection process often requires repeated interruptions during tasks, which can disrupt operators performing high-risk decision-making tasks. (2) Behavioral measurements: This approach objectively measures trust by observing and recording behavioral indicators during continuous tasks, such as acceptance of agent recommendations [13] or task performance [14]. Some studies combine behavioral measurements with self-reports [15–17] to advance quantitative trust assessment. However, behavioral indicators are often derived from specific scenarios and tasks, making it challenging to generalize their application to situational supervisory control tasks. (3) Physiological and neurophysiological measurements: These methods offer high temporal resolution, providing an alternative for real-time trust measurement [18,19]. Techniques include eye-gaze tracking [20], electrodermal activity (EDA), electroencephalography (EEG) [21], and emotional representations from voice data [22]. While these methods excel in capturing continuous data, they often struggle to explain the underlying mechanisms linking the indicators to trust. Additionally, they typically require complex and intrusive data collection equipment, which can be impractical in many applications.

In real situational supervisory control environments, operators' prominent behavioral actions involve continuously adjusting their attention allocation under high time pressure to execute high-risk decisions [23]. Evidence suggests that eye movements can intuitively reveal the context of users' activities and their cognitive processes [24]. Specific eye-tracking metrics have shown sensitivity to human trust [20], and eye-tracking devices are relatively simple to set up, causing minimal interference. This offers a promising, non-intrusive approach to achieving real-time trust monitoring. However, establishing a direct mapping between eye-tracking signals and the dynamics of operators' trust remains a complex challenge. To address this, this study aims to develop a classification and regression model based on eye-tracking data and self-report scales using machine learning algorithms. The model is designed to efficiently and cost-effectively identify operators' trust states in automated situational supervisory control tasks and is validated through simulated experiments. This research seeks to provide a novel approach for continuous, real time, and non-intrusive trust state measurement.

2. Background

This study follows Lee and See's [5] description of trust as “the attitude that an agent will help achieve personal goals in situations filled with uncertainty and vulnerability”. This definition of trust is by far the most widely used in subsequent research on trust in automation [19]. Under this definition, trust is viewed as internal information within humans, which is not directly observable [10], posing significant challenges for trust measurement. Trust is not static; it evolves dynamically over time through repeated direct and indirect interactions within a team. Increasing users' trust is insufficient; attention must also be given to the entire continuum of trust, which includes over-trust, calibrated trust, under-trust, misunderstanding, and distrust [6]. Some studies in the literature propose terms like calibration points [25] or critical states [26] to classify the situations when the intervention for calibrating trust is needed. In this study, calibrated trust is trust [that] matches the true capabilities of the automation [5], other cases lead to over-trust or under-

trust. Terms like appropriate trust, calibrated trust, and appropriate reliance have often been used interchangeably in recent research [27–29].

Trust is a continuous and dynamic psychological process [19], and self-report scales are the most intuitive tools for measuring users' psychological states. However, since self-report scales cannot provide continuous measurements, they can only capture trust levels at discrete time points. Addressing how to obtain more continuous and accurate trust dynamics is a key focus of this study. Some researchers have attempted to infer trust from the perspective of its formation and evolution. For example, Xu and Dudek [30] developed an online probabilistic trust inference model based on a dynamic Bayesian network framework, treating human trust in agents as a hidden variable that can be estimated by analyzing automation performance and agent behaviors. Similarly, Yaohui Guo [31] simulated the trust dynamics between humans and robotic agents over time, proposing a personalized trust prediction model. These mathematical models align with empirical findings on the characteristics of trust dynamics: (1) Trust at a given moment is significantly influenced by trust from the preceding moment [32]; (2) negative feedback loops with automated agents have a stronger impact on trust adjustment than positive feedback loops [32,33]; and (3) human trust tends to stabilize during repeated interactions with the same autonomous agent [34]. These studies provide a theoretical foundation and empirical insights for constructing mathematical models based on trust dynamics during actual interactions. By leveraging such models, it becomes possible to infer and fill the temporal gaps left by self-report scales, thereby enabling a more comprehensive understanding of trust dynamics.

Preliminary research has shown that raw eye-tracking data can be used to calculate various eye-tracking metrics, which have been successfully employed in past studies to infer diverse perceptual and cognitive challenges [35,36]. However, empirical evidence regarding the feasibility and validity of using eye-tracking data to infer human trust in automation remains limited. Some studies have explored the diagnostic role of a few eye-tracking metrics in specific scenarios. For instance, Lu and Sarter [20] proposed using eye-tracking metrics, such as fixation duration and saccade path length, to infer real-time trust in humans. Hergeth [37] found that gaze behavior provides a more direct measurement of trust in automation compared to other behavioral measures. Similarly, Gold C [38] demonstrated a significant negative correlation between subjective trust ratings and operators' monitoring frequency. Moreover, combining basic metrics into composite indicators [39,40] for measuring visual attention distribution offers an additional approach to assess operators' trust states more comprehensively.

In terms of algorithm selection, Lu [36] discussed the feasibility of modeling eye-tracking data using three machine learning techniques—logistic regression, k-nearest neighbors (kNN), and Random Forest—as well as two deep learning methods, MLP and CNN. The study also examined the accuracy of modeling at both individual and group levels. In practical applications, advanced deep learning models such as CNNs [41] often require significant computational resources and large datasets, which can limit their real-time applicability. Furthermore, interpretability is another critical factor influencing the adoption of these technologies. Addressing challenges such as imbalanced class distribution in training data [42] also remains an ongoing issue. These discussions provide valuable insights and references for selecting algorithmic models in this study.

3. Materials and Methods

This section focuses on detailing the participants, experimental materials, experimental tasks, data collection, machine learning models, and model evaluation. Each component is critical to the study's methodology.

3.1. Participants

A total of 12 participants were recruited from the university for this experiment. Due to interruptions during the experiment or eye tracker malfunctions resulting in incomplete

data, the data from two participants were excluded. The final analysis included data from 10 participants, aged between 19 and 28 years ($M = 23.2$, $SD = 2.6$), with 5 males and 5 females. All participants possessed theoretical knowledge of automated supervisory control tasks, with 8 of them having substantial practical experience. All participants had normal or corrected-to-normal vision and no color vision deficiencies. Written informed consent was obtained from all participants prior to data collection, and they were compensated for their participation.

3.2. Experimental Materials

A simulated system was developed specifically for this study to emulate a real-world automated supervisory control task (see Figure 1). Various targets, such as aircraft and firepower equipment, appeared and disappeared at random positions on the system's situational awareness map, following a complex yet interpretable pattern. These targets moved along predetermined paths. When the intelligent agent detected suspicious targets approaching or entering key focus areas, a popup alert would appear in the bottom-right corner of the interface.

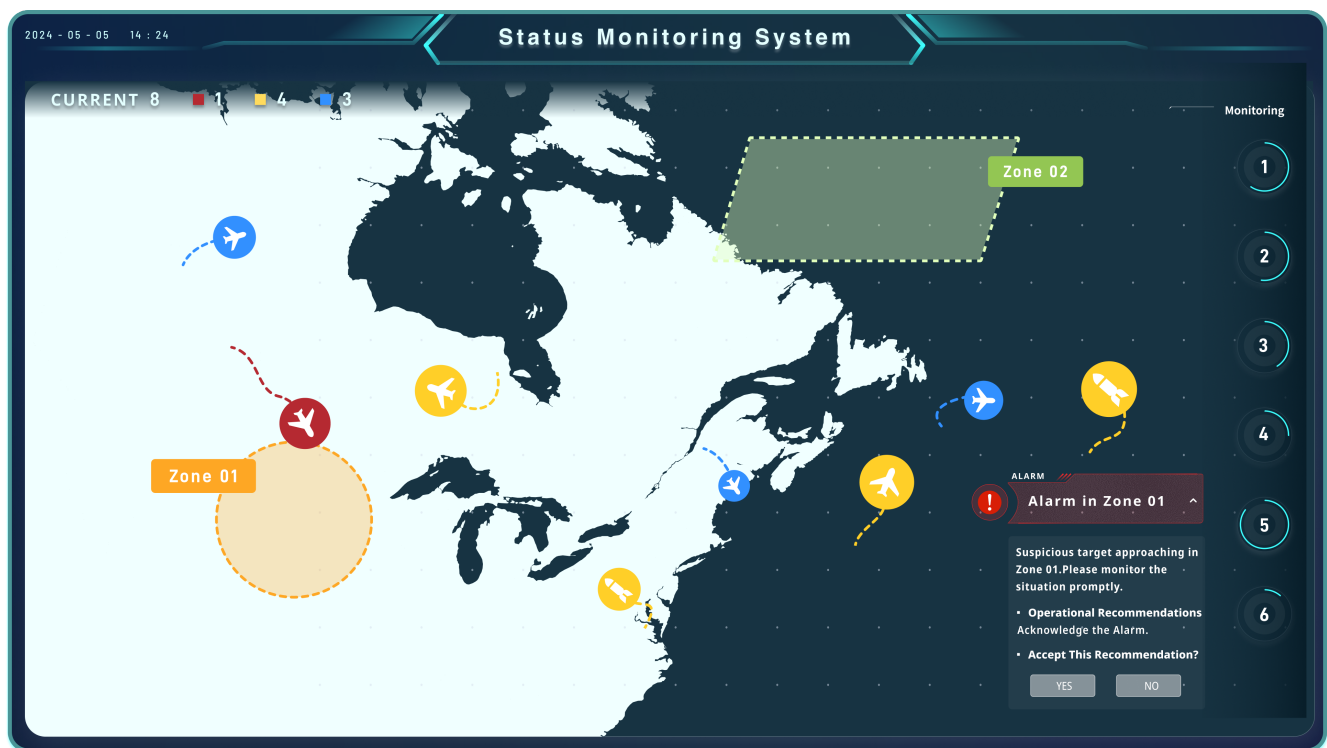


Figure 1. This is the experimental material interface at a specific moment, featuring moving targets (aircraft, fire-fighting apparatus) and hazard zones. An alert pops up in the bottom right corner when targets approach the hazard zones.

To avoid excessive cognitive load, the situational map displayed a maximum of 10 targets at any given time. Additionally, no consecutive alerts occurred within a 5 s interval to minimize overlapping popup windows, which could interfere with data collection.

The monitoring outcomes of the automated system could fall into four categories: hit, miss, false alarm, and correct rejection (see Table 1). The reliability of the agent was defined as

$$\text{Reliability} = \frac{\text{Number of Hits} + \text{Number of Correct Rejections}}{\text{Total Number of Tasks}}. \quad (1)$$

This study configured the intelligent agent in the system to operate at four predefined reliability levels: 50%, 70%, 80%, and 90%. However, due to the randomness inherent in

the experimental process, the actual reliability of the agent was calculated based on the system's backend logs after the experiment.

Table 1. The four scenarios of proxy monitoring outcomes.

True/Predicted	Alarms	No Alarms
Danger Exists	Hits	Misses
No Danger Exists	False Alarms	Correct Rejections

3.3. Experimental Tasks

Before the experiment, participants underwent a 10-minute practice session to familiarize themselves with the interface and interaction process of the simulation system. During this session, they were given a walk-through of the experimental procedure to ensure they fully understood the objectives and steps. This practice aimed to minimize the influence of individual differences, such as familiarity with the interface layout or operational habits, on the experimental results.

In the formal experimental phase, participants were instructed to supervise all appearing and disappearing targets as closely as possible. When a suspicious target approached or suddenly entered a key focus area, participants had to decide whether to accept the system's automated alert by selecting "yes" or "no" in the popup warning. This decision could be based on the situational status or their own theoretical knowledge and experience:

- Selecting "yes" prompted the system to log the alert details in the backend.
- Selecting "no" caused no system response but still logged the decision in the backend.

Each participant performed 30 continuous state discrimination tasks under each of the four system reliability levels (50%, 70%, 80%, and 90%). After every six tasks, participants completed an interim trust scale to assess their trust level at that point.

The experiment lasted approximately 40 min for each participant. Before each session, participants were reminded of the urgency and rigor required for the tasks and were informed of the system's reliability level for that session.

3.4. Data Collection

Eye-tracking data were collected using the Tobii Pro Fusion eye tracker in conjunction with the Tobii Pro Lab version 1.241 software [43], with a sampling rate of 250 Hz. A Windows 10 laptop was used for data recording and storage. Additionally, screen recordings of the experimental sessions were captured using the commercial software Bandicam version 7.1.0 [44]. The data collection setup is illustrated in Figure 2.

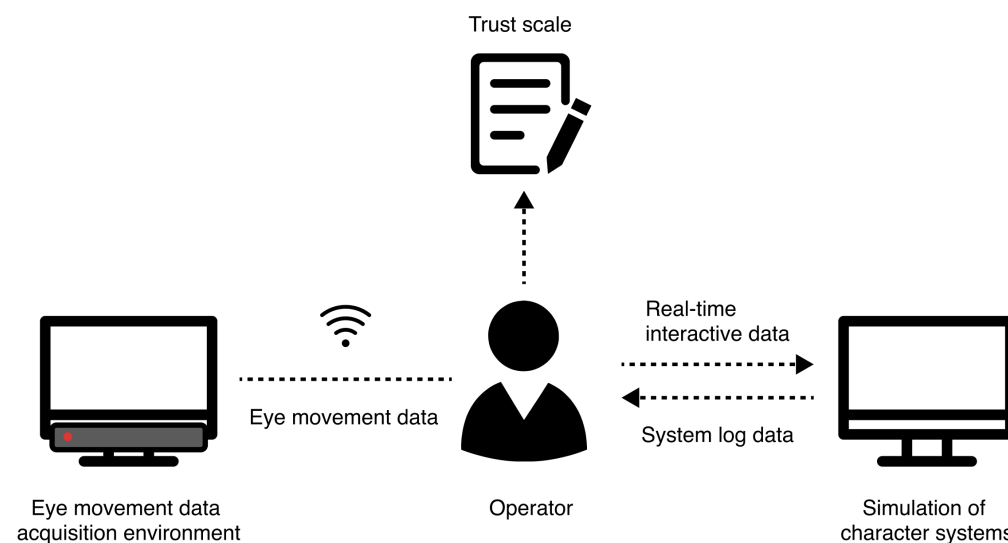


Figure 2. Human-agent trust model experimental scenario construction.

The data collected in this experiment included both objective and subjective measures, as detailed below:

A. Real-time interaction data

Real-time interaction data were gathered by observing the participants' decisions to select "yes" or "no" in response to the automated agent's alerts. These data were directly extracted from the screen recordings of the experimental sessions.

B. System log data

System log data captured the performance of the automated alert system, including the four possible outcomes: hit, miss, false alarm, and correct rejection.

C. Self-report scale data

Subjective data were obtained from participants' responses to a staged self-report trust scale administered during the experiment. The scale used was a derivative version of a validated authoritative scale [12], adapted for greater precision and relevance to participants' perceptions of the specific system used in this study.

To avoid misunderstandings due to language differences, the wording of the original items was revised appropriately to ensure a comprehensive exploration of perceived trust. Each item was rated on a 7-point Likert scale (1 = strongly disagree, 7 = strongly agree). Higher aggregate scores indicated higher levels of trust in the system. The scale's reliability and validity were tested, and the raw scores were normalized and standardized to a 100-point scale. The detailed scale items are provided in Appendix A.

D. Eye-tracking data

The eye tracker recorded various eye movement behaviors, such as fixations, saccades, and smooth pursuits, which collectively formed eye-tracking segments. To ensure comprehensive data capture, the Tobii Pro Fusion eye tracker recorded the entire experiment, including unrelated segments such as task interruptions and questionnaire completion. These irrelevant segments were trimmed using the Tobii Pro Lab software [43], resulting in two types of eye-tracking datasets:

- **Raw eye-tracking data:** This dataset included only raw fixation point coordinates represented as (x, y) pairs. Before inputting the data for analysis, preprocessing steps were applied, including handling missing data, down-sampling, and normalization. Missing data points were filled using the value from the preceding data point.
- **Feature-extracted data:** This dataset contained extracted features based on three primary eye-tracking metrics: pupil diameter, fixation, and saccade characteristics. These features were used to analyze participants' attention and trust-related behaviors.

Table 2 presents the structure of the processed data. No. represents the task code, with each sequence consisting of six tasks, repeated five times by each participant. Interaction data indicate whether the participant accepted the system's suggestion, with two possible states: yes or no. System data represent the relationship between the automated system's detection results and the actual outcomes. Scale data and eye-tracking data describe the forms in which these two types of data were collected.

Table 2. Data collection formats for objective and subjective indicators.

No.	Interaction Data	System Data	Scale Data	Eye-Tracking Data
1	No	Hit		
2	Yes	Not Hit	Complete the trust	Eye-tracker
3	No	False Alarm	scale 1 time for each	real-time collection
4	Yes	Hit	of the 6 subtasks on	of eye-tracking data
5	Yes	Correctly Reject	a scale of 1–100	throughout
6	No	Hit		

Repeat the above steps 5 times for a total of 30 completed tasks.

3.5. Trust State Classification and Regression Model Based on Two-Layer Random Forest

Based on the review in Section 2 (Background), the selected algorithm must meet the following criteria. (1) High interpretability: The model should offer strong interpretability to ensure that the relationship between input features (e.g., eye-tracking metrics) and trust states can be clearly understood and communicated. (2) Low resource usage: The algorithm should not require excessive computational resources, enabling its implementation in real-time systems without significant delays. (3) Robustness to data imbalance: The model should effectively handle imbalanced data distributions in the dataset to prevent biased predictions and ensure reliable performance across different trust states.

Random Forest [45] is a classic ensemble learning algorithm suitable for both classification and regression tasks in machine learning. The two-layer Random Forest model was chosen because it meets these requirements, achieving a balance between interpretability, computational efficiency, and adaptability to data imbalance through its ensemble learning approach and feature selection mechanisms. Additionally, the model leverages the hierarchical nature of the algorithm to enhance detection accuracy.

(1) First Layer: Random Forest Classification Model

The first layer consists of a Random Forest classification model built from multiple decision trees. These trees are trained following two fundamental principles: feature randomness and data randomness. Each decision tree uses a random subset of the raw eye-tracking data samples for training, ensuring robustness against overfitting and enhancing the model's generalization ability.

As a preliminary screening model, the first layer prioritizes the detection of significant deviations in trust levels, specifically over-trust and under-trust relative to calibrated trust. Additionally, it determines the importance ranking of each eye-tracking feature, providing crucial information for the second-layer training process and ensuring that the most relevant features are emphasized in subsequent analysis.

(2) Second Layer: Random Forest Regression Model

The second layer focuses on the most critical features identified in the first layer to enhance detection performance. This layer is trained on a refined feature set that includes the highest ranking features from the first layer, along with newly introduced metrics as input features for the regression model.

The training process constructs decision trees that leverage this curated feature set to improve accuracy and robustness. For classification tasks, the ensemble decision is made through a majority voting mechanism, where the influence of individual incorrect predictions on the final classification is minimized. For regression tasks, the results of all decision trees are averaged to produce the final regression output. Each tree contributes equally to the final outcome, and the importance of the number of weak learners (trees) in the model is secondary to the performance of individual weak learners.

The hierarchical structure of the two-layer model enhances both robustness and accuracy, providing a scalable solution for trust state detection. Figure 3 illustrates the architecture of the dual-layer Random Forest model, outlining the training process and hierarchical structure. Both parts of the Random Forest model were implemented using the standard tools provided in the Python library Scikit-learn [46].

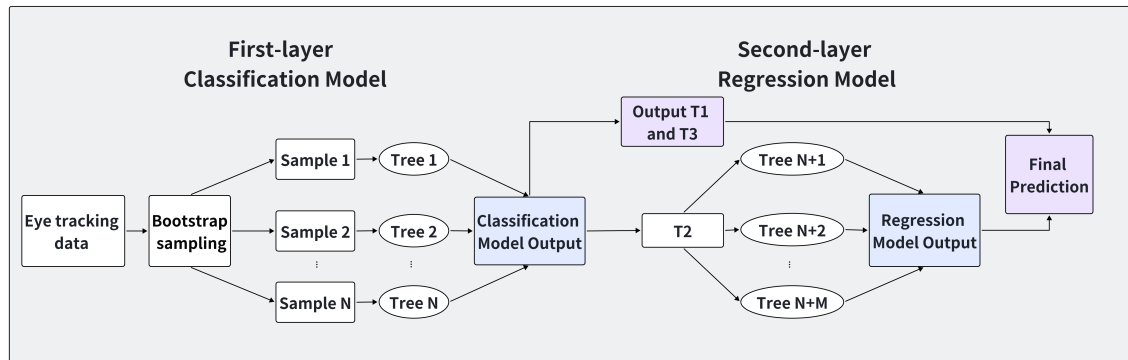


Figure 3. The architecture of the double-layer Random Forest model.

3.6. Model Evaluation

Various metrics are used to comprehensively evaluate the performance of the two-layer Random Forest model.

3.6.1. Confusion Matrix

The confusion matrix [47] is a commonly used tool for evaluating classification problems. For a K-class classification problem, it represents the classifier's prediction results in a $k \times k$ matrix format. In binary classification, samples are categorized based on the combination of their true outcomes and the model's predicted outcomes into four categories:

- True positive (TP): The model correctly predicts a positive instance.
- True negative (TN): The model correctly predicts a negative instance.
- False positive (FP): The model incorrectly predicts a negative instance as positive.
- False negative (FN): The model incorrectly predicts a positive instance as negative.

The first letter indicates whether the prediction is true (T) or false (F), while the second letter denotes the classifier's decision: positive (P) for a positive prediction and negative (N) for a negative prediction. Table 3 illustrates the types of combinations between actual and predicted outcomes.

This structure allows for detailed analysis of the model's performance, highlighting areas such as precision, recall, and overall classification accuracy.

Table 3. Types of combinations between actual and predicted outcomes.

True/Predicted	T1	T2	T3
T1	TP_1	FP_1	FP_2
T2	FN_1	TP_2	FP_3
T3	FN_2	FN_3	TP_3

3.6.2. First-Layer Random Forest Classification Model Evaluation

The first layer of the Random Forest model is a multi-class classification model with three categories. The primary metrics for evaluating the model's effectiveness are accuracy, precision, and recall [48]:

Definitions:

1. Accuracy: The proportion of correctly predicted instances out of the total instances. It evaluates the model's overall classification performance.

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN}. \quad (2)$$

2. Precision: The proportion of true positive predictions among all positive predictions made by the model. A higher precision indicates stronger ability to distinguish negative samples.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}. \quad (3)$$

3. Recall: The proportion of actual positive samples that were correctly predicted by the model. A higher recall indicates stronger ability to detect positive samples.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}. \quad (4)$$

3.6.3. Second-Layer Random Forest Regression Model Evaluation

The second layer of the Random Forest model addresses a regression task. The following common metrics were used to evaluate the model's accuracy and overall performance:

1. RSS, MSE, RMSE, and MAE provide measures of accuracy, indicating how closely the model's predictions align with the actual data.

- RSS (residual sum of squares): Measures the total squared difference between predicted and actual values.

$$\text{RSS} = \sum_{i=1}^n (y_i - \hat{y}_i)^2. \quad (5)$$

where y_i is the actual value, $\hat{y}_i$ is the predicted value, and n is the number of observations.

- MSE (mean squared error): The average of the squared differences between predicted and actual values. It penalizes larger errors more heavily.

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2. \quad (6)$$

- RMSE (root mean squared error): The square root of MSE, providing an interpretable error measure in the same units as the target variable.

$$\text{RMSE} = \sqrt{\text{MSE}}. \quad (7)$$

- MAE (mean absolute error): The average of the absolute differences between predicted and actual values. It gives equal weight to all errors.

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|. \quad (8)$$

2. R^2 reflects the model's ability to explain the variance in the data comprehensively. An R^2 value closer to 1 indicates better performance, signifying that the model captures more information from the actual labels.

- R^2 (coefficient of determination): Measures the proportion of the variance in the dependent variable that is predictable from the independent variables. It evaluates the model's ability to capture the variability in the data,

$$R^2 = 1 - \frac{\text{RSS}}{\text{TSS}}. \quad (9)$$

where TSS is the total sum of squares:

$$\text{TSS} = \sum_{i=1}^n (y_i - \bar{y})^2. \quad (10)$$

4. Data Computation and Analysis

This section provides a detailed explanation of the construction and computation of the dynamic trust mathematical model, trust state annotation, feature engineering, and hyperparameter tuning of the classifier. Each step plays a vital role in building a robust framework for trust calibration and enhancing the model's performance.

4.1. Construction and Computation of the Dynamic Trust Mathematical Model

Since self-report scales cannot provide continuous measurements, they only capture trust states at discrete time points. A key focus of this study is to obtain a more continuous and accurate representation of dynamic trust as a mapping target for eye-tracking data. Based on the review in Section 2 (Background), constructing a mathematical model of trust dynamics from actual interaction processes and using it to infer the temporal gaps in self-report scale data is theoretically feasible.

This section draws on the approach by Guo et al. [31], employing the mathematical expectation of a Beta distribution with performance-inducing parameters to model dynamic trust in the context of human–agent interactions during supervisory control tasks. The Beta distribution is particularly sensitive to the relationship between the mean and standard deviation, offering high interpretability, which makes it well suited for measuring the error between predicted and actual trust values [49].

After the agent completes the i th task, $i = 1, 2, \dots$, the human agent's temporal trust follows a Beta distribution:

$$t_i \sim \text{Beta}(\alpha_i, \beta_i). \quad (11)$$

The predicted trust $\hat{t}_i$ is calculated by the mathematical expectation of t_i :

$$\hat{t}_i = E(t_i) = \frac{\alpha_i}{\alpha_i + \beta_i}. \quad (12)$$

The parameters α_i and β_i are the defining parameters of the Beta distribution, and their update rules adhere to the principles of trust formation and evolution in human–agent interactions. (1) Influence of prior trust: Trust at the current time step is significantly influenced by the trust level at the previous time step [32]. (2) Negative feedback loops have a greater impact on trust adjustment compared to positive feedback loops [32,33]. (3) Stabilization through repeated interaction: Human trust tends to stabilize during repeated interactions with the same automated agent [34].

$$\alpha_i = \begin{cases} \alpha_{i-1} + w^s, & \text{if } r_i = 1, \\ \alpha_{i-1}, & \text{if } r_i = 0, \end{cases} \quad (13)$$

$$\beta_i = \begin{cases} \beta_{i-1} + w^f, & \text{if } r_i = 0, \\ \beta_{i-1}, & \text{if } r_i = 1. \end{cases}$$

The variable r_i represents the interaction result of the human–agent team during the i th task and can be determined by comparing the participant's real-time interaction data with the system's log data. When a positive feedback loop occurs (i.e., the participant observes that the monitoring intelligent agent issued a correct alert), $r_i = 1$. When a negative feedback loop occurs (i.e., the participant observes that the monitoring intelligent agent failed to issue a correct alert), $r_i = 0$. w_s and w_f are the gains due to the human agent's positive and negative experiences with the agent. The superscript s stands for success and f stands for failure.

After n tasks, the human–agent team accomplishes n^s tasks and fails n^f tasks. Then,

$$t_i \sim \text{Beta}(\alpha_0 + n^s w^s, \beta_0 + n^f w^f). \quad (14)$$

To infer the model's parameters, after the m th task, $m = 1, 2, \dots$, and given the human–agent interaction result $R_i = \{r_1, r_2, \dots, r_m\}$, this study determines trust $T_i = \{t_1, t_2, \dots, t_m\}$ by the parameter set $\theta = \{\alpha_0, \beta_0, w_s, w_f\}$. θ is updated with each new interaction between the human and the agent. Specifically, θ can be inferred from the trust value of the previous task and the interaction outcome of the current task.

The true value of the parameter θ is estimated as the value that maximizes the probability distribution $P(\theta)$, making it the maximum likelihood estimation (MLE) of the parameter:

$$\begin{aligned}\theta &= \underset{\theta}{\operatorname{argmax}} P(T \mid \theta, R), \\ &= \underset{\theta}{\operatorname{argmax}} \prod_{i=1}^n \operatorname{Beta}(t_i; \alpha_i, \beta_i).\end{aligned}\quad (15)$$

where α_i and β_i are determined by Equation (13).

Using the data with a reliability level of 90% as an example, the probability $P(\theta)$ can be estimated using Equation (15). Figure 4 demonstrates the prior distributions of the four parameters learned through maximum likelihood estimation (MLE).

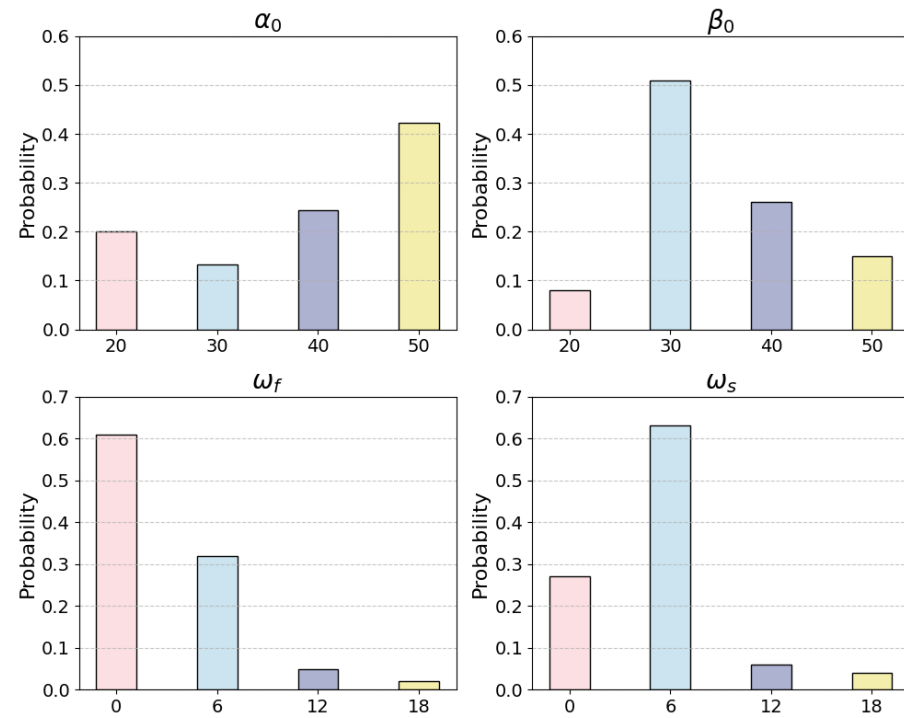


Figure 4. Prior distribution graphs of four parameters.

The proposed model successfully captured the trust dynamics of many participants, achieving high-quality data fitting. Figure 5 illustrates the prediction results for 10 participants, with the horizontal axis representing the task sequence and the vertical axis representing the trust level. The bar chart displays the trust values collected from the staged self-report questionnaires (scaled to two decimal places for computational convenience), while the line graph represents the trust levels predicted by the mathematical model. This visualization highlights the alignment between the subjective trust scores and the model's predictions, demonstrating the effectiveness of the proposed approach in estimating dynamic trust states.

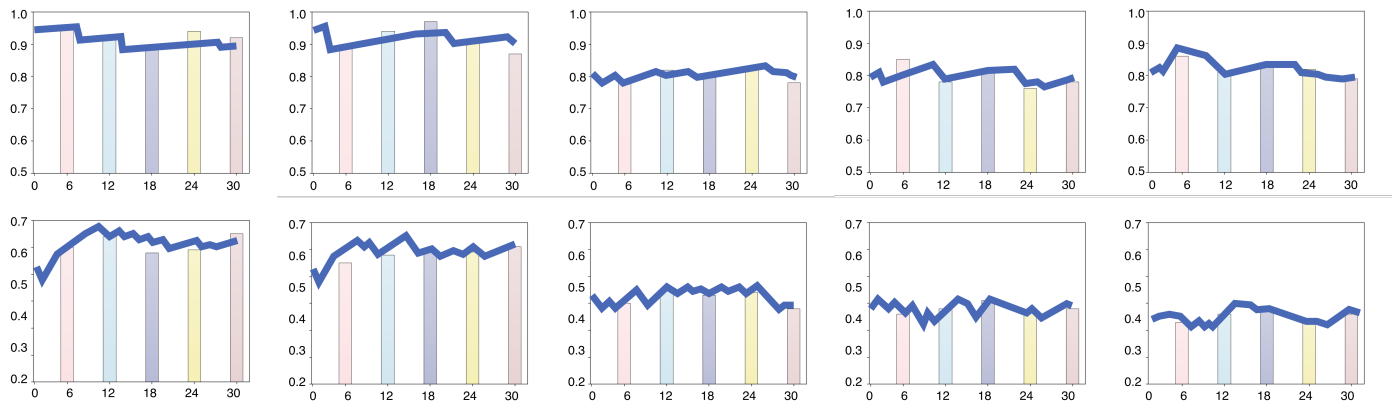


Figure 5. Prediction results of trust levels for 10 participants.

The root mean squared error (RMSE) was used to measure the deviation between the predicted values and the actual self-report scores. A smaller RMSE indicates that the predicted values are closer to the actual values, reflecting higher precision of the predictive model. In this study, the RMSE was calculated using the Python library Scikit-learn [46], and the result was 0.0521, demonstrating the good predictive performance of the model. Additionally, the trust dynamics prediction model exhibited varying performance under different system reliability levels (50%, 70%, 80%), as shown in Table 4. However, overall, the model maintained a balanced performance across these reliability levels, further validating its robustness and adaptability.

Table 4. RMSE metrics across models of varying reliability levels.

Reliability Level	RMSE
50%	0.0367
70%	0.0226
80%	0.0678
90%	0.0664
Comprehensive	0.0521

4.2. Trust State Annotation

Based on the research reviewed in Section 2 (Background), calibrated trust is trust [that] matches the true capabilities of the automation [5]; other cases lead to over-trust or under-trust. Since perceived trust and system capability are measured on different scales, comparing them is a challenge.

This section uses the model described in Section 4.1 to fill in the missing data from the self-report trust scales over time. Through normalization, the user's perceived trust scale s is expressed with a score ranging from 0 to 100. At the same time, this study measure the real reliability r of the intelligent agent based on four possible outcomes (hit, miss, false alarm, and correct rejection). The reliability r is calculated as the ratio of the number of hits and correct rejections to the total number of tasks, which is then transformed into a score ranging from 0 to 100. By comparing the mathematical relationship between s (perceived trust) and r (system reliability), the trust state can be classified into three categories:

$$\begin{cases} \text{under-trust} : s < r, \\ \text{calibrated trust} : s = r, \\ \text{over-trust} : s > r. \end{cases} \quad (16)$$

Defining intermediate trust states is challenging. To minimize the impact of imbalanced categories [42] on model accuracy and reduce noise, the distribution of data across trust categories was carefully analyzed. The boundaries for calibrated trust were expanded

to include more data under this label. As a result, trust states were classified into the following categories:

$$\begin{cases} T1 : \text{under-trust} : & s < 0.95r, \\ T2 : \text{calibrated trust} : & 0.95r \leq s \leq 1.05r, \\ T3 : \text{over-trust} : & s > 1.05r. \end{cases} \quad (17)$$

4.3. Feature Engineering

Due to the large number of eye-tracking metrics, extracting, transforming, and selecting the most valuable features is essential to improve model accuracy and enhance its generalization ability. These steps enable the model to better capture patterns within the data. Based on the studies reviewed in Section 2 (Background), preliminary research has revealed that raw eye-tracking data can be used to compute a variety of metrics. These metrics have been successfully applied in previous studies to infer various perceptual and cognitive challenges, including fixation, saccade, and pupil diameter, among others.

In this section, Spearman correlation coefficients [50] are used to examine the relationships between the three trust states annotated in Section 4.2 and the statistical metrics derived from the corresponding eye-tracking segments. The Spearman coefficient measures the direction and strength of a monotonic relationship between two variables. The correlations were calculated using the SPSS version 25.0 software [51], and the analysis results are summarized in Table 5.

Although correlation does not imply causation, the results indicate the following:

1. Most eye-tracking metrics can be used to evaluate human-machine trust states.
2. Metrics such as fixation index, saccade duration, fixation-saccade time ratio, and pupil diameter change rate show a very strong correlation with human-agent trust states.

Table 5. Eye-tracking correlation analysis.

No.	Eye-Tracking Metric	Spearman Coefficient	Correlation Result
1	Fixation Duration Mean/Fixation Duration Standard Deviation	0.803	Strong correlation
2	Fixation Index	0.853	Strong correlation
3	Horizontal Search Range	0.663	Moderate correlation
4	Vertical Search Range	0.255	Weak correlation
5	Fixation Transition Time	0.859	Strong correlation
6	Fixation-Saccade Time Ratio	0.809	Strong correlation
7	Saccade Velocity Range	0.496	Moderate correlation
8	Saccade Velocity Variance	0.542	Moderate correlation
9	Saccade Velocity Mean	0.492	Moderate correlation
10	Pupil Diameter Change Rate	0.916	Strong correlation

4.4. Hyperparameter Configuration for the Classifier

(1) Hyperparameter Selection for the First-Layer Random Forest Classification Model

The main goal at this stage is to distinguish between the T1 (under-trust) and T3 (over-trust) trust states. The eye-tracking data classified into the T2 (ambiguous trust) state is processed separately. Therefore, the overall accuracy of the model does not fully reflect the classification objective. In this section, the precision of the T1 (under-trust) and T3 (over-trust) segments is used as the criterion for selecting hyperparameters and comparing model performance.

As the number of decision trees increases, both the precision for T1 and T3 and the overall accuracy initially increase, then decrease. When the number of decision trees reaches 40, the precision for T1 and T3 and the overall accuracy all reach their optimal values. Further increasing the number of decision trees has minimal effect on the model's classification performance and may even lead to overfitting. Based on this analysis, the hyperparameter for the number of decision trees in the classification model is set to 40, with the training

and testing dataset split at a ratio of 4:1. The specific classification results are shown in Figure 6.

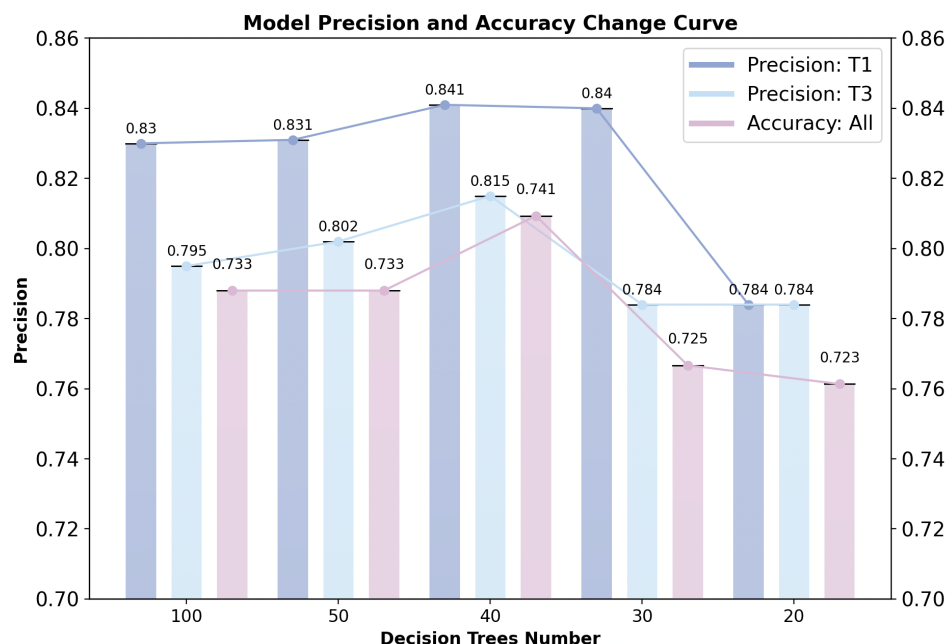


Figure 6. The changes in precision and comprehensive accuracy of T1 and T3 with the variation in the number of decision trees.

(2) Hyperparameter Selection for the Second-Layer Random Forest Regression Model

In the Random Forest regression model, the R^2 evaluation parameter and the MSE error curve change with the number of decision trees, as shown in Figure 7.

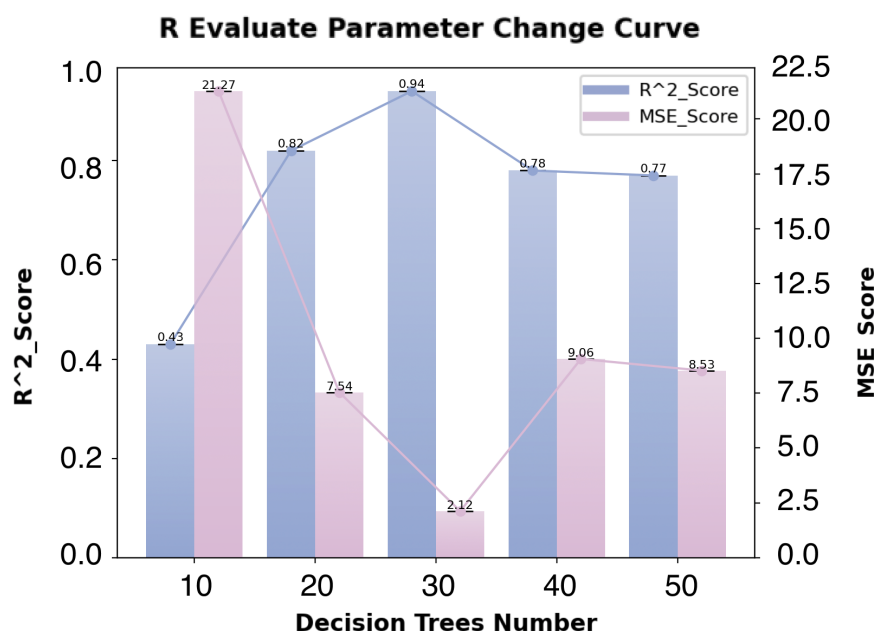


Figure 7. Change in R^2 and MSE error curve with number of decision trees in Random Forest regression.

As the number of decision trees increases from 10 to 30, the MSE error converges rapidly. When the number of decision trees reaches 30, the regression model's convergence effect slows down, and the MSE and R^2 scores are optimized. Beyond this point, increasing the number of decision trees has little effect on improving model performance. Both MSE and R^2 remain relatively stable, and the quality of the model does not improve significantly.

Therefore, the optimal hyperparameter for the number of decision trees in the regression model is set to 30, as it results in the best balance of performance for MSE and R^2 .

5. Results

This section presents the results of the first-layer Random Forest classification model and the second-layer Random Forest regression model.

5.1. First-Layer Random Forest Classification Model

(1) Classification Results

The average precision for the classification of the T1 (under-trust) and T3 (over-trust) segments is 82%. The recall rate for T1 is higher than its precision, indicating that the model performs better at identifying negative samples compared to positive samples. This suggests that the model's accuracy in correctly identifying negative trust states (T1) exceeds its ability to correctly identify positive trust states (T3). The specific results are shown in Table 6.

Table 6. Human–agent trust discrimination model evaluation results.

Evaluation Metric	Results
T1 Precision	0.80
T3 Precision	0.84
T1 Recall	0.88
T3 Recall	0.51
T1 as T3 Error Rate	0.0000
T3 as T1 Error Rate	0.0006

Based on the confusion matrix analysis in Figure 8, the classification model demonstrates the best performance on T1 segments, with the amount of data misclassified into T3 segments being less than 0.0001 and into T2 segments being 0.12, resulting in a T1 recall rate of 88%. The model also performs well on T3 segments, with the amount of data misclassified into T1 segments being less than 0.001 and the amount classified correctly into T3 segments being 0.49, resulting in a T3 recall rate of 51%.

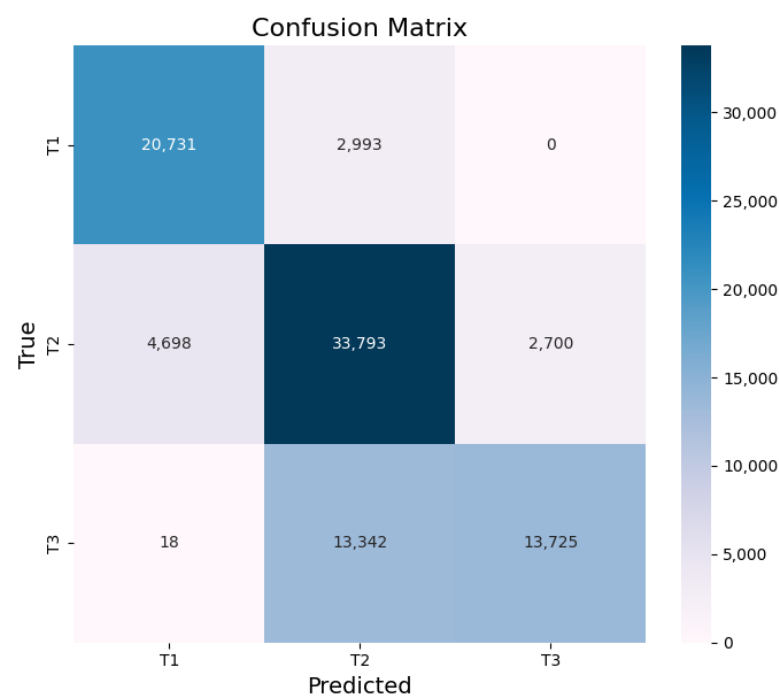


Figure 8. Confusion matrix analysis of model classification results.

In summary, the classification model demonstrates excellent performance in distinguishing T1 and T3 segments, with an overall error rate of less than 0.1%. However, the model shows less distinction between T2 and T3 segments, resulting in a relatively low T3 recall rate. Subsequently, the eye-tracking data classified as T2 segments undergoes secondary fitting using the regression model.

(2) Importance of Eye-Tracking Metrics

The importance of nine eye-tracking features was evaluated using the feature_importance method of the RandomForestRegressor, and their rankings are illustrated in Figure 9.

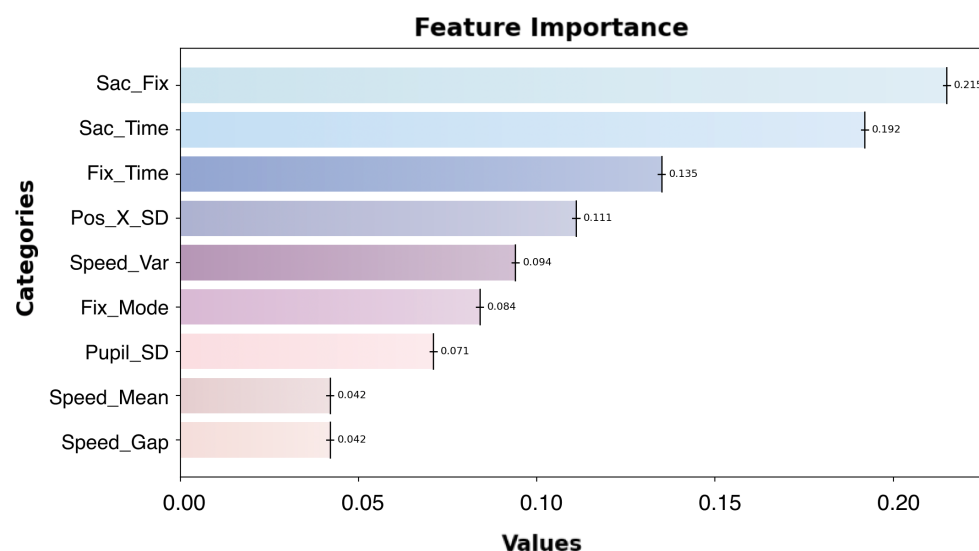


Figure 9. The importance ranking of the 9 oculomotor features output by the first-layer Random Forest classification model.

The summary of eye-tracking metrics is shown in Table 7. The features with the highest contribution are *Sac_Fix* (saccade–fixation ratio) and *Sac_Time* (saccade duration), with a combined importance exceeding 0.41. These metrics play a critical role in distinguishing between T1 (under-trust) and T3 (over-trust) states, highlighting their significance in trust calibration.

Table 7. Eye-tracking metric summary.

Eye-Tracking Metric	Description
Sac_Fix (saccade–fixation ratio)	Proportion of time spent on saccades relative to total time spent on saccades and fixations.
Sac_Time (saccade time)	Total time spent on saccadic eye movements.
Fix_Time (fixation time)	Time spent fixing on specific areas of interest (AOIs).
Pos_X_SD (horizontal gaze dispersion)	Standard deviation of gaze positions along the horizontal axis.
Speed_Var (saccade velocity variance)	Variability in the velocity of saccadic eye movements.
Fix_Mode (fixation mode)	Most frequent fixation duration observed during a task.
Pupil_SD (pupil diameter standard deviation)	Standard deviation of pupil size changes during the task.
Speed_Gap (saccade velocity range)	Range between the maximum and minimum velocities of saccadic movements.
Speed_Mean (saccade velocity mean)	Average velocity of saccadic movements.

The next most influential features are *Fix_Time* (fixation duration) and *Pos_X_SD* (horizontal gaze dispersion), which contribute notably to the model's performance. Fixation duration reflects focused attention, while horizontal gaze dispersion captures the breadth of horizontal search behavior, both of which are vital for understanding trust dynamics.

Following these are *Speed_Var* (saccade velocity variance), *Fix_Mode* (fixation mode), and *Pupil_SD* (pupil diameter standard deviation).

Finally, the features with the least impact are *Speed_Gap* (saccade velocity range) and *Speed_Var* (saccade velocity variance), each contributing less than 0.05, indicating a minimal role in the model's predictive capability.

5.2. Second-Layer Random Forest Regression Model

(1) Feature Importance

In this stage, T2 ambiguous trust segments were used as the dataset for the regression model. One eye-tracking feature, mean saccadic velocity, was removed, leaving seven other eye-tracking metrics. Additionally, a system reliability level metric was introduced, and the data were standardized. The contribution rates of the eye-tracking metrics in the second-layer Random Forest regression model are shown in Figure 10.

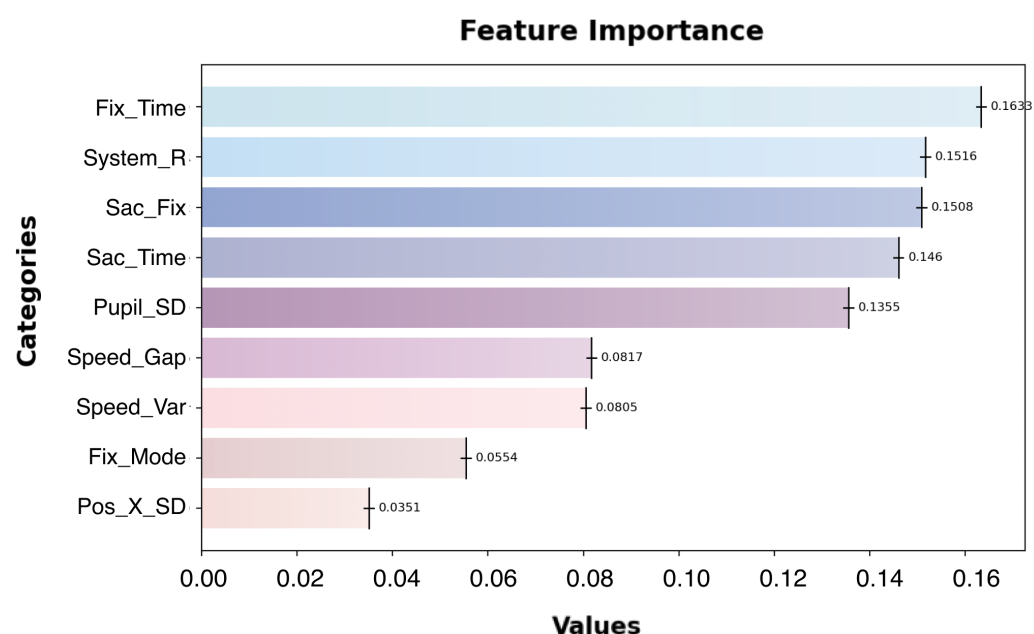


Figure 10. The importance ranking of the 9 oculomotor features output by the second-layer Random Forest regression model.

Fix_Time (fixation duration), *Sac_Time* (saccade duration), and *Sac_Fix* (saccade–fixation ratio), which are saccade–fixation-related metrics, contribute the most to trust state fitting, with a combined importance exceeding 0.45. These metrics highlight the critical role of attention dynamics in understanding and modeling trust states.

System reliability level ranks second in contribution, following the fixation and saccade metrics. This indicates that system reliability is a key environmental parameter in the evaluation and fitting of human–agent trust states, emphasizing its relevance in real-world trust calibration assessments.

(2) Model Fitting Results

The performance of the trust regression model was evaluated using the T2 segment eye-tracking dataset. The fitting results, including MSE, RMSE, and R^2 , are presented in Table 8. The MSE and RMSE values indicate good accuracy in predicting trust levels. The coefficient of determination (R^2) is 0.8993, reflecting a strong fit of the model in capturing the variance of human–machine trust states.

Table 8. The performance of the T2 fragment eye-tracking dataset within the second-layer Random Forest fitting model.

Criteria	Result
MSE	1.8439
RMSE	1.3579
R^2	0.8993

6. Discussion

This study demonstrates that eye-tracking technology can be effectively used to identify operators' different trust calibration states in automated supervisory control tasks. It highlights the correlation between eye-tracking features and human-agent trust states. Furthermore, a novel two-layer Random Forest model was introduced, establishing a trust calibration state classification and regression model based on eye-tracking features. The results showcase the high accuracy and reliability potential of eye-tracking features combined with hierarchical machine learning techniques in measuring trust calibration states.

6.1. Trust Measurement and Trust Calibration

Trust and trust calibration often coexist, and an increasing number of studies have recognized that merely increasing user trust is insufficient. It is crucial to achieve calibrated or appropriate trust levels. Measuring trust in a continuous, real time, and non-intrusive manner, and identifying whether users are in a state of over-trust or under-trust, is more beneficial for guiding system evaluation and design practices in real-world scenarios. This foundational idea serves as the starting point of our research.

Since the user-perceived scale of trust differs from the system's capability measurement, the practical implementation of trust calibration requires careful exploration. Human-agent trust was modeled using the expected value of a Beta distribution with performance-inducing parameters. Missing scores of subjective trust scales were inferred based on an array constructed from real-time interaction data and system log data within the training dataset. To represent the user's perception scale, the data were normalized and expressed on a 0–100 scale. Similarly, the system agent's reliability was measured as the ratio of the sum of hits and correct rejections to the total number of tasks, with this ratio also normalized to a 0–100 scale. Using this approach, trust states were classified into under-trust, calibrated trust, and over-trust based on the comparison between user perception and system reliability.

Currently, three well-established strategies exist for measuring trust calibration: relative measures, correlational measures, and behavioral measures. The method proposed in this study falls under correlational measures, as it directly quantifies the relationship between system capability and perceived trustworthiness. This approach offers superior real-time performance and greater flexibility across application scenarios. Unlike relative measures [52], it provides explicit diagnostics for under-trust and over-trust states. Compared to behavioral measures [53], it is more suitable for real-time assessment and does not rely heavily on predefined behavioral standards, enhancing its applicability for system design and evaluation in varied contexts.

6.2. Contribution of Eye-Tracking Metrics to Trust Calibration in Human-Agent Interaction

In the final model results, saccade duration (*Sac_Time*), fixation duration (*Fix_Time*), and saccade–fixation time ratio (*Sac_Fix*) were identified as key contributors to the determination of human-agent trust states. These metrics highlight the critical role of attention allocation in determining trust calibration states. The balance between saccades and fixations provides a window into user trust behavior, allowing systems to dynamically assess and adapt to trust states. By leveraging these insights, designers can create systems that mitigate the risks of over-trust and under-trust, fostering more effective human-agent interactions.

The saccade–fixation ratio represents the proportion of time spent on saccades relative to the total time spent on both saccades and fixations. (1) A higher ratio suggests increased exploratory behavior, often reflecting under-trust, user uncertainty, or anxiety. Users actively scan the environment and question the system’s reliability, engaging in over-monitoring behaviors. (2) A lower ratio indicates focused attention on specific targets with reduced exploratory activity, often associated with over-trust or complacency, as users rely excessively on automation and neglect active monitoring.

The saccade duration is the total time spent on saccadic eye movements (shifts between fixations). (1) A longer saccade duration indicates active monitoring behavior and is often associated with under-trust, where users frequently shift their gaze to verify system actions or gather additional information. This is common in situations involving task complexity or uncertainty about system performance. (2) A shorter saccade duration reflects over-trust, where users allocate less time to scanning the environment, assuming the system can handle tasks reliably without their intervention.

The fixation duration is the time spent fixating on specific areas of interest (AOIs). (1) A longer fixation duration indicates concentrated attention and cognitive engagement, often reflecting confidence in the system’s reliability or task-related focus. Users dedicate more time to monitoring specific elements rather than scanning widely. (2) A shorter fixation duration suggests frequent scanning or monitoring, often linked to under-trust. Users may exhibit uncertainty about the system’s reliability and compensate with increased oversight. (3) A balanced fixation duration reflects calibrated trust, where users efficiently allocate attention between confirming system performance and exploring new information.

6.3. Limitations and Future Directions

This study has certain limitations that warrant further exploration.

(1) Theoretical Scope

From a theoretical perspective, this study strictly adheres to Lee and See’s definition of calibrated trust, which is described as “trust that matches the true capabilities of automation”. However, system capability (i.e., performance) is only one of the three key factors influencing trust. In addition to performance, system processes (how the system operates internally) and purposes (the intent behind the system’s design) also play critical roles in shaping the trust process. These aspects were not fully considered in this study.

(2) Practical Complexity

In real-world applications of complex information systems, tasks are often larger in number, more diverse in type, and may exhibit non-linear relationships between them. These factors pose additional challenges to the measurement of trust. Addressing these complexities and refining trust measurement methods for such systems is a crucial direction for future research.

Moving forward, expanding the framework to incorporate system processes and purposes, and testing the proposed method in more complex, dynamic environments, will be key to advancing trust measurement and calibration research.

7. Conclusions

To address the issues of intrusiveness, subjectivity, and intermittency in current trust measurement methods, as well as the specific characteristics of automated supervisory control tasks, this study proposes a novel trust calibration state recognition method based on eye-tracking data. Key eye-tracking metrics, including saccade duration, fixation duration, and the saccade–fixation ratio, demonstrating significant contributions to human–agent trust determination. This method not only provides a new approach to measuring human–agent trust but also offers strong support for the next step in designing adaptive interfaces.

Author Contributions: Conceptualization, K.W. and W.H.; Methodology, K.W.; Software, K.W. and H.M.; Formal analysis, K.W. and L.H.; Investigation, K.W.; Data curation, H.M.; Writing—original draft, K.W.; Writing—review & editing, L.H.; Visualization, L.H.; 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: The 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-2024023.

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. Final Scale

Based on previous work on the trust scale (Gulati 2019 [12]), adaptive descriptive adjustments were made according to the system under test. The revised scale consists of the following 12 items:

- (1) I believe there could be negative consequences when using the system.
- (2) I feel I must be cautious when using the system.
- (3) It is risky to interact with the system.
- (4) I believe the system will act in my best interest.
- (5) I trust the system will do its best to assist me when I need help.
- (6) I believe the system is genuinely interested in understanding my needs and preferences.
- (7) I think the system is competent and effective in its functions.
- (8) I believe the system performs its role very well.
- (9) I trust that the system has all the functionalities I would expect from it.
- (10) If I use the system, I feel I can depend on it completely.
- (11) I can always rely on the system for support.
- (12) I trust the information presented to me by the system.

References

1. Inga, J.; Ruess, M.; Robens, J.H.; Nelius, T.; Rothfuß, S.; Kille, S.; Dahlinger, P.; Lindemann, 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. Buçinca, Z.; Malaya, M.B.; Gajos, K.Z. To trust or to think: Cognitive forcing functions can reduce overreliance on AI in AI-assisted decision-making. *Proc. ACM Hum.-Comput. Interact.* **2021**, *5*, 1–21. [\[CrossRef\]](#)
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. Wischniewski, M.; Krämer, N.; Müller, E. Measuring and understanding trust calibrations for automated systems: A survey of the state-of-the-art and future directions. In Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems, Hamburg, Germany, 23–28 April 2023; pp. 1–16.
5. Mehrotra, S.; Degachi, C.; Vereschak, O.; Jonker, C.M.; Tielman, M.L. A systematic review on fostering appropriate trust in Human-AI interaction: Trends, opportunities and challenges. *ACM J. Responsible Comput.* **2024**, *1*, 1–45. [\[CrossRef\]](#)
6. Lakkaraju, H.; Bastani, O. “How do I fool you?” manipulating user trust via misleading black box explanations. In Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society, San Jose, CA, USA, 21–23 October 2020; pp. 79–85.
7. Zhang, Z.T.; Hußmann, H. How to manage output uncertainty: Targeting the actual end user problem in interactions with AI. In Proceedings of the IUI Workshops, Virtual, 13–17 April 2021.
8. Lee, J.D.; See, K.A. Trust in automation: Designing for appropriate reliance. *Hum. Factors* **2004**, *46*, 50–80. [\[CrossRef\]](#)
9. Shin, D. The effects of explainability and causability on perception, trust, and acceptance: Implications for explainable AI. *Int. J. Hum.-Comput. Stud.* **2021**, *146*, 102551. [\[CrossRef\]](#)
10. 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.- Interact.* **2024**, *40*, 1251–1266. [\[CrossRef\]](#)
11. Razin, Y.S.; Feigh, K.M. Converging measures and an emergent model: A meta-analysis of human-automation trust questionnaires. *ACM Trans. Hum.-Robot. Interact.* **2024**, *13*, 1–14. [\[CrossRef\]](#)

12. Gulati, S.; Sousa, S.; Lamas, D. Design, development and evaluation of a human-computer trust scale. *Behav. Inf. Technol.* **2019**, *38*, 1004–1015. [[CrossRef](#)]
13. McGuirl, J.M.; Sarter, N.B. How are we doing? Presenting system confidence information to support trust calibration and adaptive function allocation. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*; SAGE Publications: Los Angeles, CA, USA, 2003; Volume 47, pp. 538–542.
14. Zhang, Y.; Liao, Q.V.; Bellamy, R.K. Effect of confidence and explanation on accuracy and trust calibration in AI-assisted decision making. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, Barcelona, Spain, 27–30 January 2020; pp. 295–305.
15. Chen, J.; Mishler, S.; Hu, B. Automation error type and methods of communicating automation reliability affect trust and performance: An empirical study in the cyber domain. *IEEE Trans. Hum.-Mach. Syst.* **2021**, *51*, 463–473. [[CrossRef](#)]
16. M. Faas, S.; Kraus, J.; Schoenhals, A.; Baumann, M. Calibrating pedestrians' trust in automated vehicles: Does an intent display in an external HMI support trust calibration and safe crossing behavior? In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, Virtual, 8–13 May 2021; pp. 1–17.
17. Albayram, Y.; Jensen, T.; Khan, M.M.H.; Fahim, M.A.A.; Buck, R.; Coman, E. Investigating the effects of (empty) promises on human-automation interaction and trust repair. In *Proceedings of the 8th International Conference on Human-Agent Interaction*, Virtual, 10–13 November 2020; pp. 6–14.
18. Barda, A.J.; Horvat, C.M.; Hochheiser, H. A qualitative research framework for the design of user-centered displays of explanations for machine learning model predictions in healthcare. *BMC Med Inform. Decis. Mak.* **2020**, *20*, 257. [[CrossRef](#)] [[PubMed](#)]
19. Gebru, B.; Zeleke, L.; Blankson, D.; Nabil, M.; Nateghi, S.; Homaifar, A.; Tunstel, E. A review on human-machine trust evaluation: Human-centric and machine-centric perspectives. *IEEE Trans. Hum.-Mach. Syst.* **2022**, *52*, 952–962. [[CrossRef](#)]
20. Lu, Y.; Sarter, N. Eye tracking: A process-oriented method for inferring trust in automation as a function of priming and system reliability. *IEEE Trans. Hum.-Mach. Syst.* **2019**, *49*, 560–568. [[CrossRef](#)]
21. Zhang, T.; Yang, J.; Chen, M.; Li, Z.; Zang, J.; Qu, X. EEG-based assessment of driver trust in automated vehicles. *Expert Syst. Appl.* **2024**, *246*, 123196. [[CrossRef](#)]
22. Deng, M.; Chen, J.; Wu, Y.; Ma, S.; Li, H.; Yang, Z.; Shen, Y. Using voice recognition to measure trust during interactions with automated vehicles. *Appl. Ergon.* **2024**, *116*, 104184. [[CrossRef](#)]
23. Wickens, C. Attention: Theory, principles, models and applications. *Int. J. Hum.-Interact.* **2021**, *37*, 403–417. [[CrossRef](#)]
24. Duan, W.; Zhou, S.; Scalia, M.J.; Yin, X.; Weng, N.; Zhang, R.; Freeman, G.; McNeese, N.; Gorman, J.; Tolston, M. Understanding the evolvement of trust over time within Human-AI teams. *Proc. ACM Hum.-Comput. Interact.* **2024**, *8*, 1–31. [[CrossRef](#)]
25. McDermott, P.L.; Brink, R.N.t. Practical guidance for evaluating calibrated trust. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*; SAGE Publications: Los Angeles, CA, USA, 2019; Volume 63, pp. 362–366.
26. Huang, S.H.; Bhatia, K.; Abbeel, P.; Dragan, A.D. Establishing appropriate trust via critical states. In *Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Madrid, Spain, 1–5 October 2018; pp. 3929–3936.
27. Bansal, G.; Bućinca, Z.; Holstein, K.; Hullman, J.; Smith-Renner, A.M.; Stumpf, S.; Wu, S. Workshop on Trust and Reliance in AI-Human Teams (TRAIT). In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, Hamburg, Germany, 23–28 April 2023; pp. 1–6.
28. Alizadeh, F.; Vereschak, O.; Pins, D.; Stevens, G.; Bailly, G.; Caramiaux, B. Building appropriate trust in Human-AI interactions. In *Proceedings of the 20th European Conference on Computer-Supported Cooperative Work (ECSCW 2022)*, Coimbra, Portugal, 27 June–1 July 2022; Volume 6.
29. Yang, F.; Huang, Z.; Scholtz, J.; Arendt, D.L. How do visual explanations foster end users' appropriate trust in machine learning? In *Proceedings of the 25th International Conference on Intelligent User Interfaces*, Cagliari, Italy, 17–20 March 2020; pp. 189–201.
30. Xu, A.; Dudek, G. Optimo: Online probabilistic trust inference model for asymmetric human-robot collaborations. In *Proceedings of the Tenth Annual ACM/IEEE International Conference on Human-robot Interaction*, Portland, OR, USA, 2–5 March 2015; pp. 221–228.
31. Guo, Y.; Yang, X.J. Modeling and predicting trust dynamics in human-robot teaming: A Bayesian inference approach. *Int. J. Soc. Robot.* **2021**, *13*, 1899–1909. [[CrossRef](#)]
32. Lee, J.; Moray, N. Trust, control strategies and allocation of function in human-machine systems. *Ergonomics* **1992**, *35*, 1243–1270. [[CrossRef](#)]
33. Manzey, D.; Reichenbach, J.; Onnasch, L. Human performance consequences of automated decision aids: The impact of degree of automation and system experience. *J. Cogn. Eng. Decis. Mak.* **2012**, *6*, 57–87. [[CrossRef](#)]
34. Yang, X.J.; Wickens, C.D.; Hölttä-Otto, K. How users adjust trust in automation: Contrast effect and hindsight bias. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*; SAGE Publications: Los Angeles, CA, USA, 2016; Volume 60, pp. 196–200.
35. Müller, R.; Reindel, D.F.; Stadtfeld, Y.D. The benefits and costs of explainable artificial intelligence in visual quality control: Evidence from fault detection performance and eye movements. *Hum. Factors Ergon. Manuf. Serv. Ind.* **2024**, *34*, 396–416. [[CrossRef](#)]
36. Lu, Y.; Sarter, N. Modeling and inferring human trust in automation based on real-time eye tracking data. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*; SAGE Publications: Los Angeles, CA, USA, 2020; Volume 64, pp. 344–348.

37. Hergeth, S.; Lorenz, L.; Vilimek, R.; Krems, J.F. Keep your scanners peeled: Gaze behavior as a measure of automation trust during highly automated driving. *Hum. Factors* **2016**, *58*, 509–519. [CrossRef] [PubMed]
38. Gold, C.; Körber, M.; Hohenberger, C.; Lechner, D.; Bengler, K. Trust in automation—before and after the experience of take-over scenarios in a highly automated vehicle. *Procedia Manuf.* **2015**, *3*, 3025–3032. [CrossRef]
39. Victor, T.W.; Tivesten, E.; Gustavsson, P.; Johansson, J.; Sangberg, F.; Ljung Aust, M. Automation expectation mismatch: Incorrect prediction despite eyes on threat and hands on wheel. *Hum. Factors* **2018**, *60*, 1095–1116. [CrossRef]
40. Parasuraman, R.; Manzey, D.H. Complacency and bias in human use of automation: An attentional integration. *Hum. Factors* **2010**, *52*, 381–410. [CrossRef]
41. Maman, Z.S.; Chen, Y.J.; Baghdadi, A.; Lombardo, S.; Cavuoto, L.A.; Megahed, F.M. A data analytic framework for physical fatigue management using wearable sensors. *Expert Syst. Appl.* **2020**, *155*, 113405. [CrossRef]
42. Li, Q.; Song, Y.; Zhang, J.; Sheng, V.S. Multiclass imbalanced learning with one-versus-one decomposition and spectral clustering. *Expert Syst. Appl.* **2020**, *147*, 113152. [CrossRef]
43. Tobii Technology. *Tobii Pro Fusion User Manual*; Technical Report; Tobii Technology: Danderyd Municipality, Sweden, 2023.
44. Bandicam Company. Bandicam Screen Recorder. 2008. Available online: <https://www.bandicam.com> (accessed on 23 May 2024).
45. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]
46. Pedregosa, F. Scikit-learn: Machine learning in python Fabian. *J. Mach. Learn. Res.* **2011**, *12*, 2825.
47. Deng, X.; Liu, Q.; Deng, Y.; Mahadevan, S. An improved method to construct basic probability assignment based on the confusion matrix for classification problem. *Inf. Sci.* **2016**, *340*, 250–261. [CrossRef]
48. Powers, D.M.W. Evaluation: From precision, recall and F-measure to ROC, informedness, markedness and correlation. *Int. J. Mach. Learn. Technol.* **2011**, *2*, 37–63.
49. McDonald, J.B.; Xu, Y.J. A generalization of the beta distribution with applications. *J. Econom.* **1995**, *66*, 133–152. [CrossRef]
50. De Winter, J.C.; Gosling, S.D.; Potter, J. Comparing the Pearson and Spearman correlation coefficients across distributions and sample sizes: A tutorial using simulations and empirical data. *Psychol. Methods* **2016**, *21*, 273. [CrossRef] [PubMed]
51. 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 17 June 2024).
52. Kraus, J.M.; Forster, Y.; Hergeth, S.; Baumann, M. Two routes to trust calibration: Effects of reliability and brand information on trust in automation. *Int. J. Mob. Hum. Comput. Interact. (IJMHCI)* **2019**, *11*, 1–17. [CrossRef]
53. Ribeiro, D.D.S.; Barbosa, G.D.J.; Silva, M.D.C.; Lopes, H.; Barbosa, S.D.J. Exploring the impact of classification probabilities on users' trust in ambiguous instances. In Proceedings of the 2021 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC), St Louis, MO, USA, 10–13 October 2021; pp. 1–9.

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.