Real-Time Detection of Robot Failures Using Gaze Dynamics in Collaborative Tasks

Ramtin Tabatabaei
The University of Melbourne
Melbourne, Australia
stabatabaeim@student.unimelb.edu.au

Vassilis Kostakos
The University of Melbourne
Melbourne, Australia
vassilis.kostakos@unimelb.edu.au

Wafa Johal
The University of Melbourne
Melbourne, Australia
wafa.johal@unimelb.edu.au

***Abstract*—Detecting robot failures during collaborative tasks is crucial for maintaining trust in human-robot interactions. This study investigates user gaze behaviour as an indicator of robot failures, utilising machine learning models to distinguish between non-failure and two types of failures: executional and decisional. Eye-tracking data were collected from 26 participants collaborating with a robot on Tangram puzzle-solving tasks. Gaze metrics, such as average gaze shift rates and the probability of gazing at specific areas of interest, were used to train machine learning classifiers, including Random Forest, AdaBoost, XGBoost, SVM, and CatBoost. The results show that Random Forest achieved 90% accuracy for detecting executional failures and 80% for decisional failures using the first 5 seconds of failure data. Real-time failure detection was evaluated by segmenting gaze data into intervals of 3, 5, and 10 seconds. These findings highlight the potential of gaze dynamics for real-time error detection in human-robot collaboration.**

***Index Terms*—Robot Failures, Gaze Dynamics, Human-Robot Collaboration, Machine Learning Classifiers**

I. Introduction

The potential for robots to assist people in various domains is becoming increasingly evident [1]–[3]. They can collaborate with humans as teammates to perform joint activities [4]. To ensure successful collaboration, it is crucial for robots to exhibit effective behaviour and communication, as this helps maintain alignment and fosters trust [5]. However, as robots become more integrated into daily life, their inevitable errors—caused by real-world uncertainties—pose risks to task success, user safety, and trust [6]–[8]. Trust in human-robot collaboration fluctuates, dropping after failures but recovering if the robot quickly detects and corrects its mistakes [9]–[11]. To recover effectively from errors, robots should not only detect their failures but also identify the specific type of failure (e.g., motion execution versus task planning). Different types of failures require specific recovery approaches [10], making accurate failure identification a key capability for robots in collaborative settings. One promising strategy for enabling robots to detect their own failures is by modelling user reactions during the moment of failure. This involves analysing signals such as social and non-verbal cues, with eye gaze emerging as a particularly valuable indicator [12]. Eye gaze conveys information about attention [13], [14], and emotional states [13], [15]. By leveraging machine learning algorithms to model user gaze behaviour, robots can monitor gaze patterns to detect failures in real-time, improving their ability to respond effectively and maintain trust.

This study explores the development of machine learning classifiers to detect robot failures using user gaze patterns during collaborative tasks. It focuses on two research questions: (RQ1) how the performance of these models varies based on the time elapsed after a robot failure, and (RQ2) how the performance of these models varies when applied to real-time failure detection.

To address these questions, we used data collected on a total of 26 participants engaged in four sessions of Tangram puzzle-solving, during which the robot was intentionally programmed to fail once per puzzle [16]. The results of the machine learning classifiers show that the models perform well in detecting failures. When implemented in real-time, they can detect most failures effectively.

II. Related Works

Research has shown that users display common instinctive social signals during robot errors, distinguishing these situations from error-free scenarios. These signals include gaze behaviour [16]–[19], facial expressions [19]–[22], verbalisation [17], [19], [20], and body movements [20], [22], [23]. For example, Peacock et al. [18] observed that gaze initially increases in motion during failures and then stabilises as users address the issue. Stiber et al. [24] identified heightened activity in facial muscles, such as smiling and brow lowering, during robot errors. Similarly, Kontogiorgos et al. [17], [19] reported increased spoken words, longer utterances, and more gaze shifts toward the robot, reflecting greater user engagement during failures.

Several studies have explored machine-learning approaches to detect failures in human-robot interactions using various behavioural and physiological cues. For example, Peacock et al. [18] trained logistic regression models on gaze dynamics to detect failures, achieving accurate detection a few seconds after the errors occurred. Similarly, Kontogiorgos et al. [19], [25] developed machine learning models, including XGBoost and Random Forest, that utilised multimodal behaviours—such as linguistic, facial, and acoustic features—to achieve high accuracy in distinguishing failure scenarios from non-failure scenarios in a verbal guidance scenario. Separately, Stiber et al. [21] trained Multi-Layer Perceptron (MLP) models using

action units (AUs) derived from facial reactions, showing that AUs are effective if users provide timely and observable responses to robot errors. Since not all users exhibit clear facial or verbal reactions to failures, this highlights a gap in the literature. This research aims to address this gap by designing classifier models based on user gaze during a collaborative task, evaluating their performance relative to the time elapsed after a robot failure, and assessing their effectiveness in real-time settings.

III. Methodology

A. Tasks Description

The experiment consisted of four collaborative tasks where a participant and a robot worked together to solve Tangram puzzles. In each task, participants were tasked with creating a unique shape using Tangram pieces. To avoid any influence of task difficulty on participant perception or behaviour, puzzles of comparable difficulty were carefully selected (See [16] for more details on the experimental setup).

Each Tangram puzzle consisted of seven pieces. The robot was responsible for placing four pieces (two small triangles, a square, and a parallelogram), while the participant placed the remaining three pieces (a medium triangle and two large triangles). The Tangram pieces were 3D-printed. The silhouettes of the puzzles were printed in black on A2-sized white paper, which was fixed to the table. Both the participant and the robot placed their pieces into the Tangram figure. Participants were instructed to move their pieces only after the robot had completed its action, with the robot always beginning by placing the first piece.

The robot's pieces were positioned near its workspace, adjacent to the paper. For each puzzle, the robot determined the placement and orientation of its pieces before initiating movement. The Tiago robot, programmed using ROS1, utilized the 'tf' library to map the pose of each piece to the coordinate frame of its robotic arm. It then employed inverse kinematics to precisely position its arm for accurate placement of each piece.

B. Robot Failures

We designed the robot to deliberately fail during each task, with failures categorized into two types: Executional Failure (EF) and Decisional Failure (DF), representing common technical issues robots may face in interactions.

- **EF:** The robot pauses for 15 seconds after grasping an object, holding it in its end effector during the pause. After the delay, it resumes the task and completes the pick-and-place action.
- **DF:** The robot picks up an object but mistakenly moves to the location intended for a different object. It places the object incorrectly, pauses for 5 seconds, and then corrects the mistake by lifting the object and placing it in the correct location.

During both failure and non-failure events, the robot followed the same pick-and-place procedure, with failures differing only in task duration: EFs added a 15-second pause, while DFs added 16.5 seconds due to incorrect placement and correction.

To avoid timing biases, the robot's malfunctions were programmed to occur at different points in each task. Specifically, failures could occur either at the beginning of the collaboration, while the robot was placing the first piece, or toward the end of the interaction, while the robot was placing the third piece.

C. Experiment

A total of 26 participants (16 females, 9 males, and 1 non-binary; aged 18–34) were recruited from a university platform. They provided informed consent, received gift vouchers as compensation, and were debriefed after the experiment.

Participants were guided by an experimenter who explained the tasks and intervened as needed to ensure safety or trigger the robot's responses. Participants wore eye-tracking glasses to record their gaze data.

Each participant completed four puzzles, with each puzzle lasting approximately 3 minutes, followed by a short break between puzzles. The study was conducted in a controlled laboratory setting with participants who had no prior experience with robotics. In each puzzle, the robot was responsible for placing four pieces, correctly placing three and making a failure with one. This failure was pre-programmed to vary by type (Executional or Decisional) and timing (Early or Late). These combinations were counterbalanced using a four-condition Latin-Square design, ensuring balanced exposure across conditions and minimizing timing effects.

Participant ID	Puzzle 1	Puzzle 2	Puzzle 3	Puzzle 4
1	EF (Early)	EF (Late)	DF (Late)	DF (Early)
2	EF (Late)	DF (Early)	EF (Early)	DF (Late)
3	DF (Early)	DF (Late)	EF (Late)	EF (Early)
4	DF (Late)	EF (Early)	DF (Early)	EF (Late)
5	EF (Early)	EF (Late)	DF (Late)	DF (Early)
...	...	...	...	...
14	EF (Early)	EF (Late)	DF (Late)	DF (Early)
...	...	...	...	...

TABLE I: Order of failure type with their timing across puzzles

D. Measures

For each puzzle and piece, the robot's actions (e.g., moving or picking up objects), failure occurrences, and failure types were recorded, along with participant gaze data collected using Neon Eye Tracking Glasses. For more details on the methodology, refer to Paper [16].

Using the gaze data, we calculated several metrics, including the average rate of gaze shifts towards all AOIs, the average rate of gaze shifts towards the robot's body, the average duration of gaze directed at the end effector, the probability of gazing at each AOI, transition entropy, and stationary entropy. For successful pickups and placements, metrics were calculated from when the robot began picking up the object until it placed the piece. For EFs, metrics covered the 15-second failure period, while for DFs, they spanned from the robot's movement towards the incorrect location to completing the placement and pausing for 5 seconds.

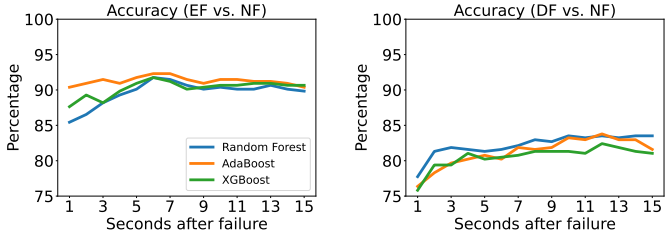
Using gaze behaviour metrics, machine learning models were trained to classify each type of failure against a no-failure condition in a binary manner. Each participant contributed 12 data points for non-failure conditions, 2 data points for EFs, and 2 data points for DFs. Five classifiers were employed: Random Forest, configured with 100 decision trees; AdaBoost, with 100 boosting iterations; XGBoost, performing 100 boosting rounds with a learning rate of 0.01; Support Vector Machine (SVM), using a linear kernel; and CatBoost, configured with 100 iterations, a learning rate of 0.1, and a tree depth of 6. Classifiers were implemented using Scikit-learn, XGBoost, and CatBoost libraries.

To address data imbalance and prevent bias, we applied SMOTE normalization with k-neighbors set to 2. The trained models were evaluated using two approaches. First, we assessed their performance in distinguishing failure events from non-failure events based on the first n seconds of a failure. Second, to evaluate real-time failure detection, we analysed the models' performance in identifying failure types by segmenting the eye-tracking data into intervals of 3, 5, and 10 seconds, using a sliding window of 1 second. For both evaluation methods, we employed leave-one-out cross-validation, where models were trained on data from 25 participants and tested on the data from the remaining participant.

The primary goal of the classifiers was to achieve high accuracy while minimising false negatives, as failing to detect a failure event is critical in this context. Given this aim, we focus on reporting only the accuracy and recall metrics.

IV. RESULTS

A. Evaluating Classification Performance with Varying Failure Times



(a) (b)

Fig. 1: Classifier accuracy assessed using the first n seconds of the failure period: a) distinguishing between non-failure and executional failure, and b) distinguishing between non-failure and decisional failure.

To address the first research question, we evaluated the trained models (Random Forest, AdaBoost, and XGBoost), which were trained on the entire duration of non-failure and failure periods. We assessed their performance using only the first n seconds of the failure period. Figure 1 illustrates the average accuracy of the models in distinguishing between non-failure (NF) and executional failure (EF), as well as between non-failure and decisional failure (DF). As n increases to 5 seconds, the accuracy stabilizes. For distinguishing EF from NF, the accuracy remains around 90%, with a Recall of Failure of approximately 94% across all classifiers. Similarly, for distinguishing DF from NF, the accuracy stabilises around 80%, with a Recall of Failure of approximately 90% for all classifiers.

B. Evaluating Classification Performance for Real-Time Failure Detection

To enable the robot to detect its mistakes in real-life scenarios, it needs to repeatedly check at regular intervals whether something has gone wrong. In this section, we aim to address both research questions. In addition to the machine learning models used in the previous section, we also include SVM and CatBoost here.

The models were trained on the entire duration of non-failure and failure periods and evaluated by segmenting the eye-tracking metrics data into intervals of 3, 5, and 10 seconds, using a sliding window of 1 second. Figures 2a, 2b, 2c, and 2d illustrate the performance of the models in distinguishing between NF and EF and between NF and DF.

For both NF vs. EF and NF vs. DF, Random Forest achieved the highest accuracy compared to other classifiers, with an accuracy of approximately 60%. In contrast, SVM had the lowest accuracy, below 50%. However, SVM was the most effective in detecting the highest number of failures for both failure types.

Additionally, we calculated the percentage of users, during each 3-, 5-, and 10-second interval of the failure phase, for whom the model successfully detected the failure. Figure 3a shows these percentages for EF at the 5-second interval, and Figure 3b shows them for DF at the same interval.

The results showed that, for EF, the models were most accurate at detecting user reactions during the period from 4 to 7 seconds for the 3-second interval, and from 3 to 8 seconds for the 5-second interval after a failure began. Similarly, for DF, the optimal detection period occurred from 2 to 5 seconds for the 3-second interval, and from 1 to 6 seconds for the 5-second interval after a failure began.

V. DISCUSSION AND CONCLUSION

This study highlights the importance of user gaze dynamics in detecting robot failures during collaborative tasks. The results demonstrate that gaze-based machine learning classifiers can identify robot errors with high accuracy when the models are trained by labelling each pick-and-place action as either a failure or a non-failure, and tested similarly while reducing the duration of failure periods. However, since the exact moment of a robot failure is unknown, the robot needs to repeatedly analyse user gaze at regular intervals to determine whether a failure has occurred. For this purpose, we tested time intervals of 3, 5, and 10 seconds. Although this real-time approach does not achieve the same level of accuracy as the previous method, it allows robots to continuously monitor for EFs or DFs. The models were more effective at distinguishing between NF and EF than between NF and DF. Among the classifiers, Random Forest achieved the highest accuracy, but its recall of failures

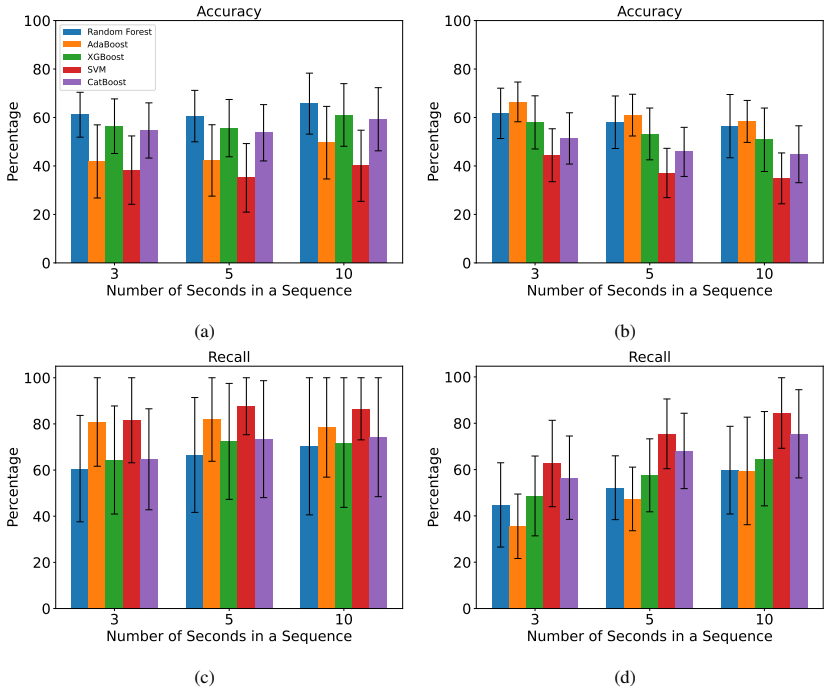


Fig. 2: Classifier performance evaluated using eye-tracking metrics segmented into intervals of 3, 5, and 10 seconds with a 1-second sliding window: a) Accuracy in distinguishing between non-failure and executional failure, b) Accuracy in distinguishing between non-failure and decisional failure, c) Recall in detecting executional failure, and d) Recall in detecting decisional failure.

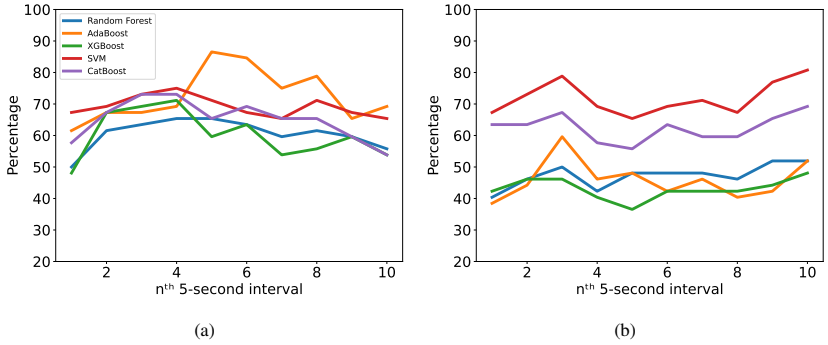


Fig. 3: Percentage of users for whom the model successfully detected failures during each 5-second interval of the failure phase: a) Executional Failure, and b) Decisional Failure.

was lower than others. For higher recall rates, SVM may be a better option as it detects the highest number of failures.

Additionally, we analysed the percentage of users whose failures were correctly detected within each 5-second interval after a failure began. For EF, this percentage was approximately 70%, while for DF, the results varied across classifiers, with CatBoost reaching around 60%. Unlike user facial expressions in response to failures, as studied in [24], gaze reactions do not exhibit specific characteristics like reaction time and reaction duration. As shown in the results, varying the duration of failure periods or using different 5-second intervals yielded consistent performance across models.

Despite these promising results, several limitations remain. In some cases, the robot's actions, such as placing its piece, overlapped with participants planning their next move, which could affect model accuracy. Furthermore, the study relied solely on gaze behaviour as an error indicator. Integrating multimodal cues, such as facial expressions, body movements, and speech, could enhance detection accuracy and robustness.

In conclusion, leveraging user gaze dynamics for robot error detection represents a significant step toward improving the reliability and trustworthiness of collaborative robots. This approach has the potential to enhance human-robot collaboration by enabling robots to proactively detect and recover from errors in real-time.

ACKNOWLEDGEMENT

This research is partially supported by the Australian Research Council Discovery Early Career Research Award (Grant No. DE210100858)

References

[1] A. Sauppé and B. Mutlu, "The Social Impact of a Robot Co-Worker in Industrial Settings," in *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, CHI '15, (New York, NY, USA), pp. 3613–3622, Association for Computing Machinery, Apr. 2015.

[2] F. Babel, A. Vogt, P. Hock, J. Kraus, F. Angerer, T. Seufert, and M. Baumann, "Step Aside! VR-Based Evaluation of Adaptive Robot Conflict Resolution Strategies for Domestic Service Robots," *International Journal of Social Robotics*, vol. 14, pp. 1239–1260, July 2022.

[3] S. Chatterjee, R. Chaudhuri, and D. Vrontis, "Usage Intention of Social Robots for Domestic Purpose: From Security, Privacy, and Legal Perspectives," *Information Systems Frontiers*, vol. 26, pp. 121–136, Feb. 2024.

[4] L. Mingyue Ma, T. Fong, M. J. Micire, Y. K. Kim, and K. Feigh, "Human-Robot Teaming: Concepts and Components for Design," in *Field and Service Robotics* (M. Hutter and R. Siegwart, eds.), (Cham), pp. 649–663, Springer International Publishing, 2018.

[5] M. Desai, M. Medvedev, M. Vázquez, S. McSheehy, S. Gadea-Omelchenko, C. Bruggeman, A. Steinfeld, and H. Yanco, "Effects of changing reliability on trust of robot systems," in *Proceedings of the seventh annual ACM/IEEE international conference on Human-Robot Interaction*, HRI '12, (New York, NY, USA), pp. 73–80, Association for Computing Machinery, Mar. 2012.

[6] K. E. Schaefer, J. Y. C. Chen, J. L. Szalma, and P. A. Hancock, "A Meta-Analysis of Factors Influencing the Development of Trust in Automation: Implications for Understanding Autonomy in Future Systems," *Human Factors: The Journal of the Human Factors and Ergonomics Society*, vol. 58, pp. 377–400, May 2016.

[7] M. Salem, G. Lakatos, F. Amirabdollahian, and K. Dautenhahn, "Would You Trust a (Faulty) Robot? Effects of Error, Task Type and Personality on Human-Robot Cooperation and Trust," in *Proceedings of the Tenth Annual ACM/IEEE International Conference on Human-Robot Interaction*, HRI '15, (New York, NY, USA), pp. 141–148, Association for Computing Machinery, Mar. 2015.

[8] S. S. Sebo, P. Krishnamurthi, and B. Scassellati, ""I Don't Believe You": Investigating the Effects of Robot Trust Violation and Repair," in *2019 14th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pp. 57–65, Mar. 2019. ISSN: 2167-2148.

[9] G. LeMasurier, A. Gautam, Z. Han, J. W. Crandall, and H. A. Yanco, "Reactive or Proactive? How Robots Should Explain Failures," in *Proceedings of the 2024 ACM/IEEE International Conference on Human-Robot Interaction*, HRI '24, (New York, NY, USA), pp. 413–422, Association for Computing Machinery, Mar. 2024.

[10] L. Wachowiak, A. Fenn, H. Kamran, A. Coles, O. Celiktutan, and G. Canal, "When Do People Want an Explanation from a Robot?," in *Proceedings of the 2024 ACM/IEEE International Conference on Human-Robot Interaction*, HRI '24, (New York, NY, USA), pp. 752–761, Association for Computing Machinery, Mar. 2024.

[11] J. M. Kraus, J. Merger, F. Gröner, and J. Pätz, "'Sorry' Says the Robot: The Tendency to Anthropomorphize and Technology Affinity Affect Trust in Repair Strategies after Error," in *Companion of the 2023 ACM/IEEE International Conference on Human-Robot Interaction*, HRI '23, (New York, NY, USA), pp. 436–441, Association for Computing Machinery, Mar. 2023.

[12] L. Wachowiak, P. Tisnikar, G. Canal, A. Coles, M. Leonetti, and O. Celiktutan, "Analysing Eye Gaze Patterns during Confusion and Errors in Human–Agent Collaborations," in *2022 31st IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*, (Napoli, Italy), pp. 224–229, IEEE, Aug. 2022.

[13] B. M. Velichkovsky, A. Kotov, N. Arinkin, L. Zaidelman, A. Zinina, and K. Kivva, "From Social Gaze to Indirect Speech Constructions: How to Induce the Impression That Your Companion Robot Is a Conscious Creature," *Applied Sciences*, vol. 11, p. 10255, Jan. 2021. Number: 21 Publisher: Multidisciplinary Digital Publishing Institute.

[14] Y. Fang, J. Tang, W. Shen, W. Shen, X. Gu, L. Song, and G. Zhai, "Dual Attention Guided Gaze Target Detection in the Wild," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11385–11394, June 2021. ISSN: 2575-7075.

[15] C.-M. Huang, S. Andrist, A. Sauppé, and B. Mutlu, "Using gaze patterns to predict task intent in collaboration," *Frontiers in Psychology*, vol. 6, July 2015. Publisher: Frontiers.

[16] R. Tabatabaei, V. Kostakos, and W. Johal, "Gazing at failure: Investigating human gaze in response to robot failure in collaborative tasks," in *Proceedings of the 2025 ACM/IEEE International Conference on Human-Robot Interaction*, HRI '25, (New York, NY, USA), Association for Computing Machinery, Mar. 2025.

[17] D. Kontogiorgos, S. van Waveren, O. Wallberg, A. Pereira, I. Leite, and J. Gustafson, "Embodiment Effects in Interactions with Failing Robots," in *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20, (New York, NY, USA), pp. 1–14, Association for Computing Machinery, Apr. 2020.

[18] C. E. Peacock, B. Lafreniere, T. Zhang, S. Santosa, H. Benko, and T. R. Jonker, "Gaze as an Indicator of Input Recognition Errors," *Proceedings of the ACM on Human-Computer Interaction*, vol. 6, pp. 142:1–142:18, May 2022.

[19] D. Kontogiorgos, M. Tran, J. Gustafson, and M. Soleymani, "A Systematic Cross-Corpus Analysis of Human Reactions to Robot Conversational Failures," in *Proceedings of the 2021 International Conference on Multimodal Interaction*, ICMI '21, (New York, NY, USA), pp. 112–120, Association for Computing Machinery, Oct. 2021.

[20] N. Mirnig, M. Giuliani, G. Stollnberger, S. Stadler, R. Buchner, and M. Tscheligi, "Impact of Robot Actions on Social Signals and Reaction Times in HRI Error Situations," in *Social Robotics* (A. Tapus, E. André, J.-C. Martin, F. Ferland, and M. Ammi, eds.), (Cham), pp. 461–471, Springer International Publishing, 2015.

[21] M. Stiber, R. Taylor, and C.-M. Huang, "Modeling Human Response to Robot Errors for Timely Error Detection," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 676–683, Oct. 2022. ISSN: 2153-0866.

[22] L. Wachowiak, P. Tisnikar, A. Coles, G. Canal, and O. Celiktutan, "A Time Series Classification Pipeline for Detecting Interaction Ruptures in HRI Based on User Reactions," in *Proceedings of the 26th International Conference on Multimodal Interaction (ICMI '24)*, ACM, Aug. 2024.

[23] P. Trung, M. Giuliani, M. Miksch, G. Stollnberger, S. Stadler, N. Mirnig, and M. Tscheligi, "Head and shoulders: automatic error detection in human-robot interaction," in *Proceedings of the 19th ACM International Conference on Multimodal Interaction*, ICMI '17, (New York, NY, USA), pp. 181–188, Association for Computing Machinery, Nov. 2017.

[24] M. Stiber, R. H. Taylor, and C.-M. Huang, "On Using Social Signals to Enable Flexible Error-Aware HRI," in *Proceedings of the 2023 ACM/IEEE International Conference on Human-Robot Interaction*, HRI '23, (New York, NY, USA), pp. 222–230, Association for Computing Machinery, Mar. 2023.

[25] D. Kontogiorgos, A. Pereira, B. Sahindal, S. van Waveren, and J. Gustafson, "Behavioural Responses to Robot Conversational Failures," in *Proceedings of the 2020 ACM/IEEE International Conference on Human-Robot Interaction*, HRI '20, (New York, NY, USA), pp. 53–62, Association for Computing Machinery, Mar. 2020.