Trusted AI Challenge – Fall Stage White Paper

Jinwoo Oh, Anirudh More, David Du, Aida Akbarzadeh, Nathan Lau

Outline:

- Problem Description
- Objectives / Metrics of Success
- Architecture
- Optimization Algorithm
- Trust
- Next Steps
- References

Problem Description:

The problem focuses on the human-AI team for traversal of unmanned ground vehicle UGV), under presence of multiple supporting agents including unmanned aerial vehicle (UAV), AI, and human operator. UGV traversal through mine-laden terrain under uncertainty of mine existence. UAV can traverse and scan the terrain in advance. Two types of agents are utilized to identify the mine existence from UAV-scanned information: AI and human operators. AI provides faster assessments but is generally known to be less reliable under certain environmental conditions. Human operators are slower but generally more accurate. With the help of these agents, UGV can traverse to the destination with less uncertainty.

Given the possibility of wrong terrain identification of UAV, AI, and human operators, discrepancy between the prediction and reality can significantly affect the trust of humans on the system. Therefore, this problem also seeks to consider trust as a substantial factor in designing and implementing the system.

Objectives / Metrics of Success:

There are two objectives overall: (1) fastest UGV arrival to the destination, while (2) securing human trust on an AI-enabled system.

First, the fastest UGV arrival to the destination can be represented as an optimization problem, minimization of traversal time of the UGV. The system can utilize UAV to scan the terrain and collect data while balancing the priority between AI and human operators for mine detection tasks. With all information combined, the UGV traverses terrain through the dynamically planned route. Therefore, the challenge lies in optimizing routes of UGV, UAV, as well as accounting for the different detection capabilities of the subsystems.

Second, building human trust in the system should be also considered. Not only time objective, which is prone to terrain misidentification and wrong arrival time, trust-affecting components such as user interface and interaction, must be considered together in overall process so that the system can provide sufficient trust to the humans (ex. commanders).

Architecture:

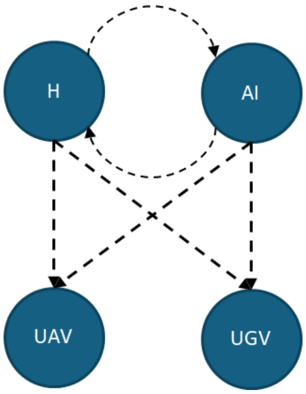


Figure 1: Multi-agent Architecture

The mine-clearing operation employs a multi-agent architecture comprising human operators, AI, UAV, and UGV working in coordinated roles, as depicted in Figure 1 above. This architecture enables dynamic role allocation between human operators and AI, supporting both autonomous operation and human oversight, ensuring that critical decisions benefit from both computational efficiency and human judgment. The interaction between agents follows a networked structure where information flows bidirectionally between human operators, AI, and unmanned vehicles, enabling real-time adaptation to changing conditions while maintaining operational coherence. This integrated system facilitates efficient mine detection and clearance while maintaining operational safety.

The architecture centers on two decision-making agents - human operators and AI - that work collaboratively to analyze terrain data and make critical mine detection decisions. The human operator provides high-accuracy assessments but requires approximately 30 minutes per terrain segment analysis. In contrast, the AI system processes data significantly faster, completing analyses in approximately one minute, though its accuracy varies based on environmental conditions including ground type, seasonal factors, topographical features, and ambient lighting conditions.

The physical components of the architecture consist of two unmanned vehicles with distinct operational roles. The UAV serves as the primary reconnaissance platform, collecting video imagery of the terrain for subsequent analysis. This aerial surveillance provides crucial data that informs subsequent ground operations. The UGV functions as the tactical implementation unit, conducting physical mine-clearing operations based on the processed intelligence from both human and AI analyses. This dual-vehicle approach enables safe distance operations while maintaining operational effectiveness.

The system implements a structured workflow for mine detection and clearance operations. Initially, both human operators and AI analyze UAV-collected imagery to identify potential mine locations. Following this analysis, they identify and prioritize links for UAV surveillance, direct the UAV to specific locations, and command its traversal patterns. Similarly, for ground operations, they identify suitable links for UGV deployment, direct the vehicle to these locations, and oversee its traversal. The system accounts for variable completion times in its operational planning. Standard link traversal requires approximately 20 minutes when no explosive devices are encountered. However, when mines are detected, the processing time extends to approximately one hour per link, reflecting the additional complexity and safety measures required for proper handling of mines.

Optimization Algorithm:

The mine-clearing mission can be conceptualized as an undirected network where each node represents a critical decision point in the minefield, and each edge represents a potential path between these points. In the operational scenario, this network could span hundreds of nodes across the mission area, with each node serving as a waypoint for vehicle routing decisions. The edges connecting these nodes represent traversable paths that need to be inspected by UAVs and potentially cleared by UGVs. The network structure allows for multiple possible routes between connected points, enabling path update (i.e. dynamic path planning) as new information about mine presence is gathered. This representation facilitates the implementation of path-planning algorithms that can help determine the safest and most time-efficient route through the minefield.

To optimize the actions of agents in the system for the timely arrival of the agents to its destination, two types of optimization algorithms will be utilized. For the mine-clearing UGV, path planning algorithms will solve the problem of identifying the shortest or fastest route between connected nodes in a network. For the scanning (UAV) and decision-making agents (AI, and human operators), algorithms for the Rural Postman Problem (RPP) will determine the most efficient way to traverse and scan key network edges while minimizing costs. By incorporating interpretable optimization algorithms, we can manipulate the factors in human-AI interaction with higher degrees of freedom and explore their relationship with the trust.

The optimization will incorporate a bi-level planning approach to enable the independence of the UAV and UGV. In particular, the algorithm will be built on the work of Bhadoriya et al. [1], who introduced a coordination strategy for UGV-UAV teams. To navigate UGV to the fastest path under uncertain impedance (i.e. traffic congestion), the original framework used the Rural Postman Problem with Time Windows (RPP-TW) on UAV to scan the terrain, terrain identification strategies for AI and human operator, and the D* algorithm for UGV to dynamically plan the optimal path under the latest information available. Adapting the framework to our challenge, the D* algorithm enables incorporating heuristic factors such as the human agents' trust for node search prioritization. Additionally, our adaptation will incorporate a new factor—prediction accuracy factor of AI and human agents—which was not considered in the original study.

The UAV, AI and human operators will ensure safety and optimality for the UGV traversal. Initially, the agents will traverse and update paths for the UGV to traverse, prioritizing adjacent edges to UGV so that these agents support the UGV to follow the best identified path in real-time.

For the route planning and scheduling of UAV, the **Rural Postman Problem with Time Windows (RPP-TW)** will be utilized as a base model [2], where a certain set of edges must be visited within specific time constraints. However, as the algorithm only suggests the case where the starting point and the destination is the same, some modifications are needed, so modified formulation provided below will be used for our challenge.

1. **Input parameters:**
    - A graph $G(V, E)$, where V denotes vertices and E denotes edges.
        - The graph is transformed into $G = (N, A)$, where N is a set of nodes representing the two possible directions of required edges in G, and A denotes set of edges connecting nodes representing shortest paths.
        - δ_i denotes a set of vertices incident to edge i.
    - A subset of edges $E_R \subseteq E$ of required edges.

    - Required edges denote the optimal path of the UGV, so that we ensure those edges are scanned by UAV and AI or human operators before UGV traversal.
- A starting point (e_s) and termination point (e_t) of the vehicle.
    - $\sum_{j\in N} x_{sj} = 1$ and $\sum_{j\in N} x_{jt} = 1$ to ensure the agent starts at s and ends at j.
- Time windows $[a_i, b_i]$ for each required edge $e_i \in E_R$.
- Cost (c_{ij}) and traversal time (T_{ij}) for edges.

2. **Objective**: Minimize the total traversal cost $\sum_{i\in A}\sum_{j\in A|\delta_j\cap\delta_i\neq 0, j\neq i}\sum_{k=1}^{m}\sum_{l=1}^{m} c_i x_{ijkl}$, which includes servicing all the required edges and traversing non-required edges within time window.
    - x_{ijkl}: Binary variable, where $x_{ijkl} = 1$ if edge $e_{ij}$is traversed for the first time in kth travel(copy).

3. **Constraints**:
    - Each required edge must be serviced within its specified time window (= [0, UGV entrance time]).
    - A vehicle can wait at an edge if it arrives earlier than the release time, but the service starts only when the time window opens.
    - The vehicle can traverse edges multiple times, with traversal costs depending on whether the edge is serviced or deadheaded.

Determining the priority of terrain identification: Since UGV traversal time is generally longer than AI identification or UAV scan times, these agents may have additional capacity to evaluate other paths. In such cases, they will assess terrain along secondary routes (e.g., the second- or third-fastest paths) to reduce map uncertainty. In this case, these agents will scan and identify the terrain on suboptimal paths (i.e. 2nd fastest path, 3rd fastest path), thereby reducing the uncertainty of the map and helping to identify a new path that is expected to be the new optimum.

While there is only one UAV which scans the terrain, two agents identify the terrain - AI and human operator, and these agents can work in parallel, and they can identify terrain wherever the UAV has scanned. This necessitates strategically 1) providing priority of each edge to determine which edge should be identified first, and 2) further consideration of parallel utilization.

For the prioritization of edge identification, we will prioritize identifying edges that change the estimated traversal time the most. If there are multiple options with the same amount of change in estimated traversal time, we will prioritize the one that changes the number of edges the most. As an example of Figure 2, mine detection at edge 1 will change the route more than point 2. In this case, we prioritize identification of edge 3.

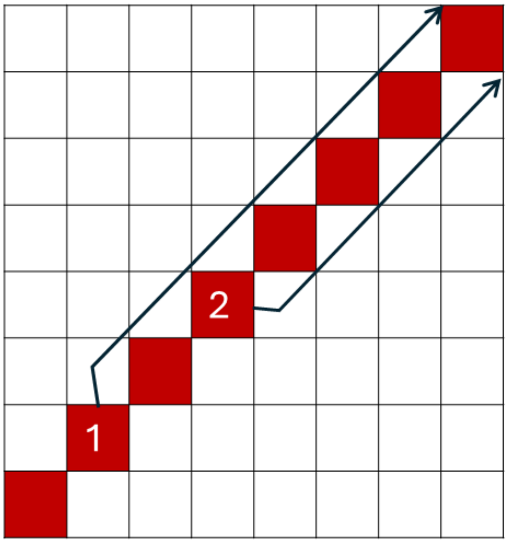


Figure 2. Prioritization of Route Identification

Parallel Utilization of AI and human operators in terrain identification: Also, given the priority of the edges, terrain identification strategies for utilizing AI and human operator should be established. To maximize the trust, we seek human operators to double-check the terrain with the AI. Therefore, we will utilize AI to analyze the entire terrain based on the priority provided above and then allocate human agents to the highest-priority edge where they can complete scanning before the UGV reaches that point. By doing so, we expect this can maximize resource utilization and human-AI trust by guiding the UGV to traverse terrain that has been scanned by the AI and verified by a human operator within the limited resources.

To configure the UGV for real-time path optimization under uncertainty, the **D* Lite Algorithm** [3] will be employed. This lightweight algorithm is specifically designed for dynamic environments where edge costs (i.e. traversal time) can change. An example is goal-directed robot navigation tasks in unknown terrains where only the origin and destination are defined, and the robot updates its path dynamically based on real-time observations (e.g., detecting unimpeded or impeded edges during traversal). For the mine-clearing missions, this dynamic adaptability is critical, as traversal costs depend on whether mines are detected along a path or not.

The algorithm starts with generating an initial path based on an initial map, using a heuristic cost estimation for unexplored edges to prioritize the exploration of the nodes considering information availability. For instance, in optimal weather conditions with clear visibility, the algorithm would provide more weight to the AI's inspection or UAV's detection result, potentially allowing faster path exploration. Conversely, in challenging conditions such as heavy rain or rough terrain, the algorithm would become more stringent, potentially requiring additional verification or choosing alternative routes. The heuristic function: $h(s, \text{goal}) = h_{\text{dist}}(s, \text{goal}) \times h_{\text{accuracy}}(s) \times t_{\text{estimated}}(s)$ estimates the cost from the current node s to the goal based on three components:

1. **Distance estimate (h_{dist})**: The shortest path length between s and the goal multiplied by t_{base} = 21 minutes for unexplored edges (1 minute UAV inspection + 20 minutes UGV) and 20 minutes for explored edges).
2. **Prediction accuracy factor (h_{accuracy})**: A multiplier $1 + \sum_{i=1}^{4} w_i\, f_i(s)$, where f_i (s) represents environmental factors such as ground type, season, topography, and lighting, and w_i represents the weighting for each factor.
3. **Estimated time ($t_{\text{estimated}}(s)$)**: Accounts for mine probability p_{mine}, calculated as $t_{\text{base}} \times (1 + p_{\text{mine}} \times \Delta t_{\text{mine}})$, where Δt_{mine} = 40 minutes is the additional clearance time for mine-affected paths ($t_{base} = 20$ minutes for traversal without mine, $t_{base} + \Delta t_{mine} = 60$ minutes for traversal with mine existence).

As agents (e.g., UAVs) provide updated information about edge costs (e.g., mine presence), the algorithm efficiently reflects these updates in real-time recalculating only affected nodes in the map. The map is represented as an undirected cyclic graph G = (V, E), where V is the set of nodes (passage points), and E is the set of edges (paths). Each edge has an associated cost:

$$c(u,v) = t_{\text{UAV}} + t_{\text{UGV}},$$

where:

- t_{UAV} = 0 minute or 1 minute depending on if the edge was inspected.
- t_{UGV} = 0 minute if edge was not traversed, 20 minutes for clear path traversal, or 60 minutes if mines are detected.

The algorithm maintains two key values for each node:

1. **g(s):** The current best-known cost to reach node s.
2. **rhs(s):** A one-step lookahead value estimating the minimum cost to reach node s.

The priority of nodes in the search is determined using a **key function**:

$$k(s) = [k_1(s), k_2(s)]$$

where:

- $k_1(s) = \min(g(s), rhs(s)) + h(s_{start}, s) + k_m$
- $k_2(s) = \min(g(s), rhs(s))$

Here, k_m is an accumulated cost adjustment based on changes in edge costs due to updated trust or mine detection data.

The main loop of D* Lite proceeds as follows:

1. Extract the node with the smallest key value from the priority queue.
2. Update its values:
   - If $g(s) > rhs(s)$: Set $g(s) = rhs(s)$.
   - Otherwise: Set $g(s) = \infty$.
3. Propagate changes to neighboring nodes by updating their *rhs* values:
   - For each neighbor *v*:

$$rhs(v) = \begin{cases} 0, & v = s_{goal} \\ min_{u \in Succ(v)}(c(v,u) + g(u)), & v \neq s_{goal} \end{cases}$$

This process ensures that only affected portions of the graph are recalculated when edge costs change due to new observations or trust updates.

Trust can also be integrated into the D* Lite algorithm by modifying its heuristic function to include trust-based cost estimations. The algorithm's heuristic function, which traditionally estimates the cost of moving between nodes, can be augmented with a trust component that influences path selection based on the reliability of AI predictions [4]. This ensures that paths with higher trust values are preferred while maintaining admissibility ($h(s,s') \leq c*(s,s')$), where $c*(s,s')$ includes both distance and trust metrics. The trust-weighted heuristic would prioritize paths where AI predictions align with human assessments, effectively guiding the search toward routes with higher trust values. When edge costs change due to trust updates, D* Lite recomputes only the affected portions of the network. This approach ensures that the UGV not only finds efficient paths but also traverses routes that maintain appropriate levels of human trust in the system's decision-making process.

Trust:

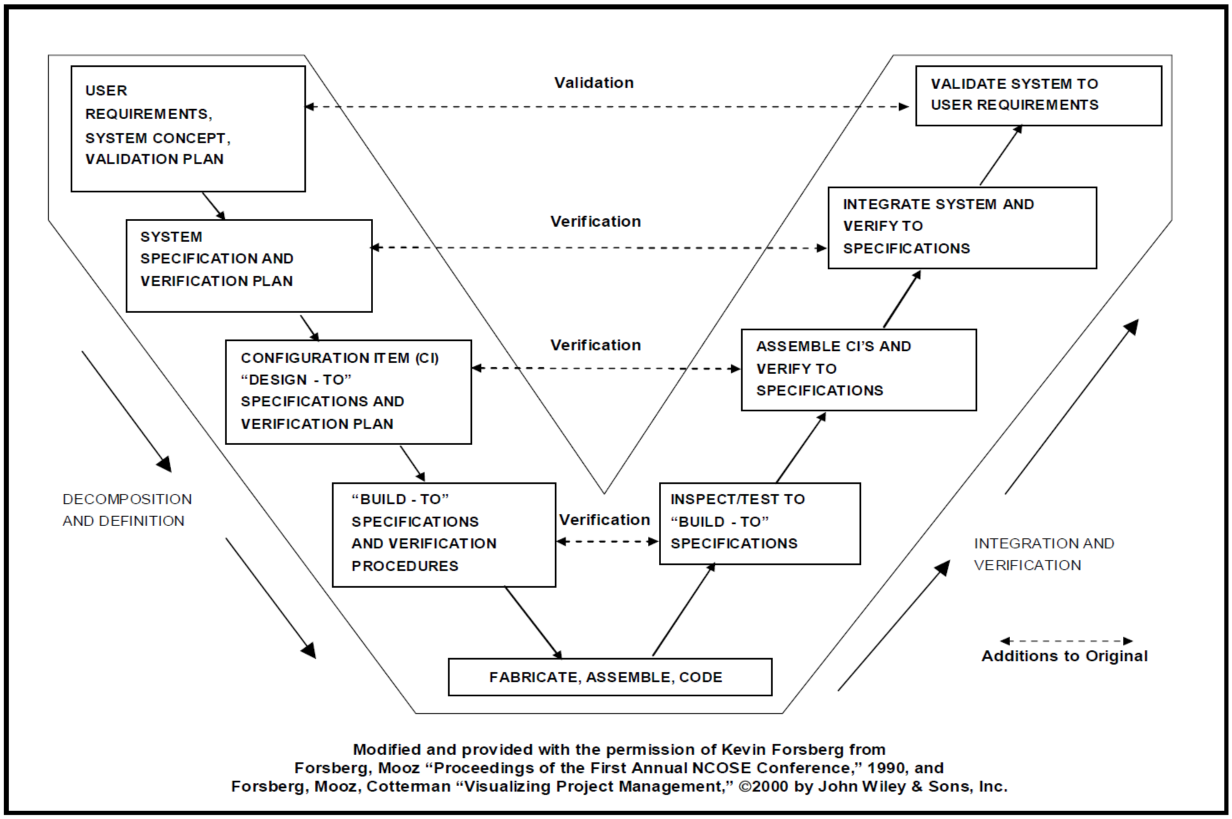
Trust in AI-enabled systems represents a critical dimension of system effectiveness, particularly in safety-critical environments. At a fundamental level, trust acts as a mediator between system capabilities and operator decision-making, impacting whether operators choose to rely on, question, or override AI outputs. Without appropriate levels of trust, operators may either dismiss valuable AI insights—thereby losing potential efficiency gains—or accept AI-driven recommendations uncritically, which heightens the risk of catastrophic errors. Trust can be influenced by a system's inherent reliability [5] [6], the clarity with which it communicates uncertainty [7], and the ability of human operators to understand, anticipate, and verify the AI's reasoning and outputs [8]. As such, trust is not a static attribute; it is a dynamic construct that evolves through repeated interactions, iterative improvements in system design, and ongoing operator training.

In the specific case of the mine-clearing mission, the importance of trust in AI systems is amplified by the high stakes and inherent uncertainty of the operating environment. Aerial reconnaissance by the UAV offers critical terrain data, but the mined landscape, varying environmental conditions, and imperfect sensor feeds introduce uncertainty into terrain classification and subsequent risk assessments. The complexity of the conditions means that both human operators and the AI may misidentify mined areas. Here, the AI offers speed but may falter under certain environmental conditions, while human operators, though generally more accurate, are slower.

Trust becomes essential as operators weigh AI-driven rapid assessments against their own analyses. If operators lack confidence in the AI's predictions, they may discard valuable time-saving recommendations, causing delays and potentially exposing ground assets to prolonged risk. Conversely, if operators trust the AI blindly—even when conditions degrade its accuracy—they risk directing the UGV into hazardous terrain. Thus, calibrating trust levels is key to achieving both mission's main objectives: minimizing UGV traversal time and maintaining overall safety. Success in this scenario depends on enabling operators to understand when and how to rely on AI, ensuring that any trust placed in the system is informed, appropriate, and adaptive.

Rationale for an Extended V-Model

To better address these trust-related challenges, it is useful to consider how traditional Systems Engineering frameworks, like the Department of Defense (DoD) V-model in Figure 3, focus primarily on defining operational needs, deriving technical requirements, and verifying and validating system performance. While effective in managing technical complexity and ensuring mission-critical reliability, the conventional V-model is inherently limited in capturing the human-centric aspects that shape trust in human-AI collaborations. As discussed in earlier sections of this work, emerging complexities in multi-agent, AI-enabled systems—such as opacity, unpredictability, and the dynamic evolution of trust—remain insufficiently addressed by these traditional approaches.


USER REQUIREMENTS, SYSTEM CONCEPT, VALIDATION PLAN
Validation
VALIDATE SYSTEM TO USER REQUIREMENTS
SYSTEM SPECIFICATION AND VERIFICATION PLAN
Verification
INTEGRATE SYSTEM AND VERIFY TO SPECIFICATIONS
CONFIGURATION ITEM (CI) "DESIGN - TO" SPECIFICATIONS AND VERIFICATION PLAN
Verification
ASSEMBLE CI'S AND VERIFY TO SPECIFICATIONS
DECOMPOSITION AND DEFINITION
"BUILD - TO" SPECIFICATIONS AND VERIFICATION PROCEDURES
Verification
INSPECT/TEST TO "BUILD - TO" SPECIFICATIONS
INTEGRATION AND VERIFICATION
Additions to Original
FABRICATE, ASSEMBLE, CODE

Modified and provided with the permission of Kevin Forsberg from
Forsberg, Mooz "Proceedings of the First Annual NCOSE Conference," 1990, and
Forsberg, Mooz, Cotterman "Visualizing Project Management," ©2000 by John Wiley & Sons, Inc.

Figure 3: Original V-Model

To address this gap, we propose an integrated approach that merges SE and Human-Systems Integration (HSI) principles, extending the DoD V-model to embed trust-focused HSI activities throughout the system lifecycle. By treating human operators and their interactions with AI as integral elements of the system rather than peripheral concerns, the extended V-model encourages human-centered AI considerations—such as transparency, interpretability, and support for trust calibration—to be integrated into early design decisions. In doing so, it acknowledges that technical proficiency alone does not suffice in safety-critical, multi-agent environments like the mine-clearing mission. Instead, a balanced, trust-informed perspective is needed to ensure that human-AI teams can reliably adapt, coordinate, and make sound decisions under uncertainty.

Applying the Extended V-Model to Foster Trust

Within the mine-clearing scenario, the extended V-model would allow for defining trust-related requirements early in the Stakeholder Requirements Definition and Requirements Analysis phases. For instance, trust metrics—such as operator confidence thresholds, acceptable frequencies of AI misclassification, or the degree of transparency in AI decision-making—can be established alongside performance and time objectives. Subsequent design and implementation stages can then incorporate these trust metrics into interface design, decision-support tools, and operator training protocols. Verification and validation activities can be conducted through human-in-the-loop simulations that replicate mine-laden terrain, allowing early assessment of whether the system fosters correct calibration of trust and identifying where adjustments to algorithms, interfaces, or training materials might be needed.

Suggested Activities and Artifacts to Build and Validate Trust

Specific HSI activities can further operationalize trust-building within the extended V-model:

- **Trust Requirements Specification:**
  Introduce explicit trust requirements that define acceptable rates of AI misclassification under various environmental conditions, thresholds for operator reliance, and criteria for explainability. These specifications should guide the design of AI algorithms, user interfaces, and decision-support aids.
- **Participatory Design Workshops:**
  Involve mission commanders and field operators early on in participatory design activities. These workshops help identify operator expectations regarding transparency, explainability, and confidence-building features in the AI system’s interfaces and outputs, ensuring that trust considerations guide initial design.
- **XAI-Enhanced User Interface Prototypes:**
  Develop and iteratively test interfaces that visualize confidence levels, highlight environmental factors influencing accuracy, and explain AI-driven recommendations, helping operators gauge when and how to rely on the AI’s assessments.
- **Scenario-Based Training and Simulation Exercises:**
  Conduct training sessions and virtual simulations replicating uncertain conditions—e.g., lighting, terrain, or ambiguous sensor data—to allow operators to experience system capabilities firsthand. These simulations enable operators to develop accurate mental models of the AI’s performance, refine their trust levels, and understand the system’s limit.
- **Human-in-the-Loop Verification and Validation:**
  Involve operators directly in test scenarios that assess the AI’s mine detection performance. By observing how operator trust evolves in response to AI-generated assessments over multiple trials, engineers can quantify the impact of design modifications on trust and guide iterative improvements.

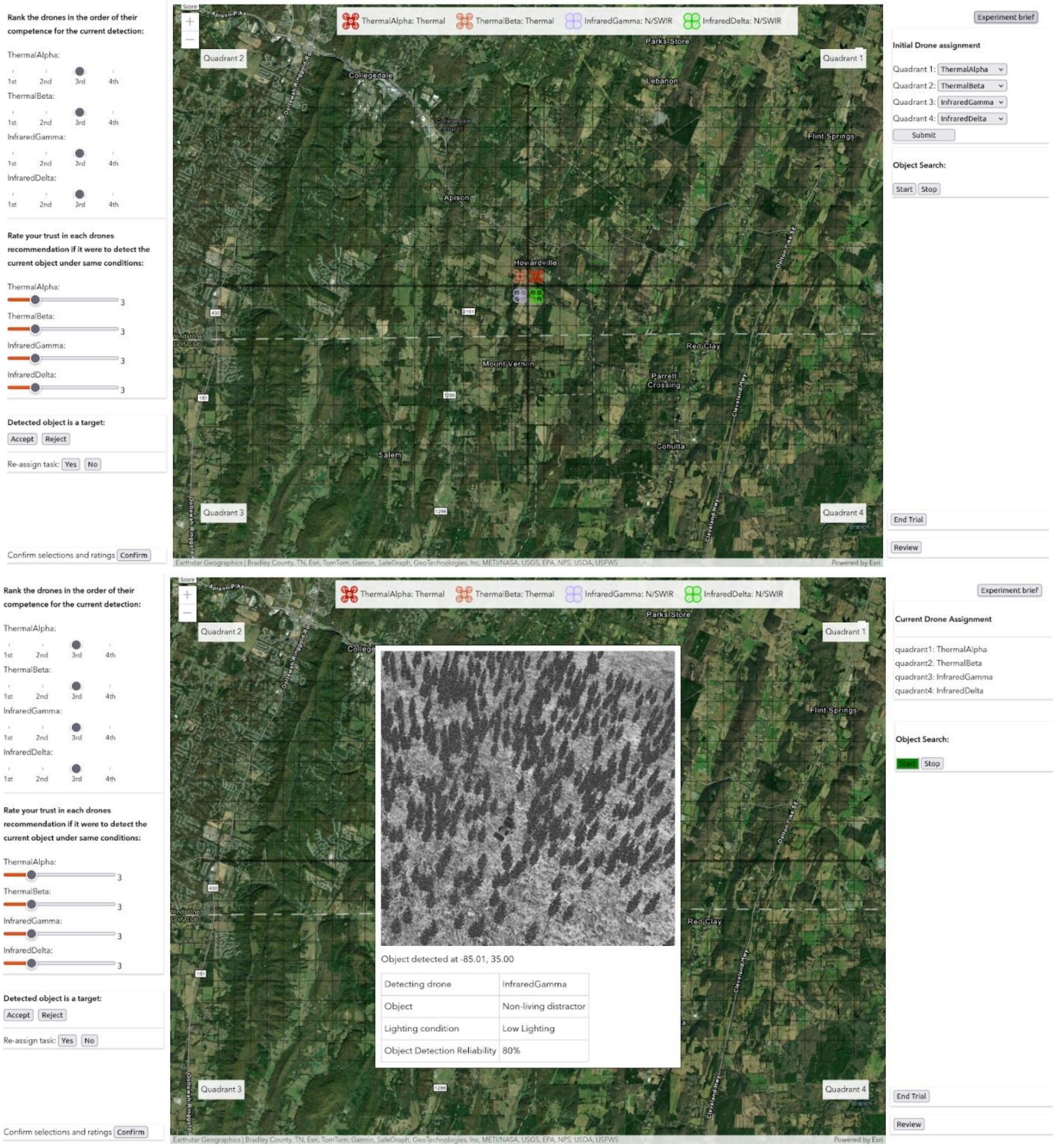
- **Continuous Feedback Mechanisms and Trust Monitoring:**
  Incorporate trust assessment tools—such as post-task questionnaires, physiological measures, or interaction logs—to monitor fluctuations in operator trust over time. These metrics enable ongoing refinement of algorithms, interfaces, and operator support materials, which ensures the system adapts to both mission demands and evolving user trust profiles.

By systematically embedding these trust-focused measures and HSI principles into the development lifecycle, the extended V-model helps ensure that the mine-clearing mission benefits from both the speed of AI-enabled assessments and the informed judgment of human operators.

Next Steps:

User-interactive components will be designed to test effects of various design components on human trust, to derive the best possible user interaction design. Specifically, a customized program will be configured to support investigating effects of various components on human trust, such as fidelity, amount of information and details provided, and communication mode. We expect these prototype implementations will allow us to observe human behavior and trust over various user interfaces and map conditions.

We plan to leverage a working testbed already configured that can examine human confidence and trust on AI in a search-and-rescue problem under human-AI teaming (Figure 4), with multiple simulated UAVs searching over defined areas. The differences between the testbed and the given problem are (1) there is only one UAV in this challenge, and (2) there is an UGV configured to traverse with the shortest path. We plan to accommodate these differences and derive an interface design optimized for this challenge and test their effect on human trust.



Rank the drones in the order of their competence for the current detection:

ThermalAlpha: 1st 2nd ●3rd 4th

ThermalBeta: 1st 2nd ●3rd 4th

InfraredGamma: 1st 2nd ●3rd 4th

InfraredDelta: 1st 2nd ●3rd 4th

Rate your trust in each drones recommendation if it were to detect the current object under same conditions:

ThermalAlpha: 3

ThermalBeta: 3

InfraredGamma: 3

InfraredDelta: 3

Detected object is a target:

Initial Drone assignment

Quadrant 1: ThermalAlpha

Quadrant 2: ThermalBeta

Quadrant 3: InfraredGamma

Quadrant 4: InfraredDelta

Object Search:

Current Drone Assignment

quadrant1: ThermalAlpha
quadrant2: ThermalBeta
quadrant3: InfraredGamma
quadrant4: InfraredDelta

Object detected at -85.01, 35.00

Detecting drone	InfraredGamma
Object	Non-living distractor
Lighting condition	Low Lighting
Object Detection Reliability	80%

Figure 4: Testbed of Human-AI integration in Search-and-Rescue Problem

Additionally, we will consider how the allocation of tasks between human operators and AI agents affects trust dynamics. In the current effort, we have assumed a sequential approach, where only one agent—human operator or AI—is assigned a specific task (e.g., terrain estimation) at a time. As we move forward, we plan to investigate parallel and redundant parallel modes of task allocation. In a parallel mode, the AI and human operator could simultaneously focus on different terrain segments, potentially enabling faster coverage but also introducing different trust calibration challenges as operators weigh distinct streams of information. In the redundant parallel approach, both the human and AI might examine the same areas, providing opportunities for cross-verification but also raising questions about how conflicting assessments influence operator trust. By exploring these different modes of collaboration, we aim to understand not only how to optimize task assignments for efficiency and accuracy but also how trust evolves under varying degrees of redundancy and concurrency.

References:

[1] Bhadoriya, A. S., Rathinam, S., Darbha, S., Casbeer, D. W., & Manyam, S. G. (2023). Assisted Path Planning for a UGV-UAV Team Through a Stochastic Network. arXiv preprint arXiv:2312.17340.

[2] Monroy-Licht, M., Amaya, C. A., & Langevin, A. (2014). The rural postman problem with time windows. Networks, 64(3), 169-180.

[3] Koenig, S., & Likhachev, M. (2002, July). D* lite. In Eighteenth national conference on Artificial intelligence (pp. 476-483).

[4] Ferguson, D., Likhachev, M., & Stentz, A. (2005, June). A guide to heuristic-based path planning. In Proceedings of the international workshop on planning under uncertainty for autonomous systems, international conference on automated planning and scheduling (ICAPS) (pp. 9-18).

[5] Kaplan, A. D., Kessler, T. T., Brill, J. C., & Hancock, P. A. (2023). Trust in Artificial Intelligence: Meta-Analytic Findings. Human Factors, 65(2), 337-359.

[6] Schaefer, K.E. (2016). Measuring Trust in Human Robot Interactions: Development of the “Trust Perception Scale-HRI”. In: Mittu, R., Sofge, D., Wagner, A., Lawless, W. (eds) Robust Intelligence and Trust in Autonomous Systems. Springer, Boston, MA.

[7] Chen, Jessie & Lakhmani, Shan & Stowers, Kimberly & Selkowitz, Anthony & Wright, Julia & Barnes, Michael. (2018). Situation awareness-based agent transparency and human-autonomy teaming effectiveness. Theoretical Issues in Ergonomics Science. 19. 259-282. 10.1080/1463922X.2017.1315750.

[8] Shih-Yi, Chien., Michael, Lewis., Katia, Sycara., Jyi-Shane, Liu., Asiye, Kumru. (2018). The Effect of Culture on Trust in Automation: Reliability and Workload. 8(4):29-. doi: 10.1145/3230736