

Cooperative Intelligent Control of Multi-Agent Systems (MAS) Through Communication, Trust, and Reliability

Stefalo Acha¹, Sun Yi^{2,*}

^{1,2} Department of Mechanical Engineering, North Carolina A&T State University, Greensboro NC, USA

Email: ¹ sacha@aggies.ncat.edu, ² syi@ncat.edu

*Corresponding Author

Abstract—The field of Multi-Agent Systems (MAS) has achieved significant advancements in modern research and development. This study focuses on enhancing trust evaluation, communication efficiency, and adaptive navigation in scenarios where agents have limited prior knowledge. Key contributions include the development of a high-intelligence MAS system that integrates key input data, such as real-time parameters regarding agents' distances from one another, their distances to target locations, weather conditions, visibility, machine learning capabilities, positions relative to safe or unsafe environments for trust evaluation, delays in communication, and potential cyber threats. These factors trigger a dynamic topology-switching mechanism to secure agents or minimize communication delays in high-security operations. The MAS implements these strategies based on an adaptive communication model, enabling agents to execute various steps during data pooling effectively. Agents utilize real-time data to coordinate flock movements, ensuring dynamic and robust control through data pooling. For example, in a topology requiring a lead agent, the lead agent provides navigation instructions based on pooled data collected during mission execution. This data may involve repositioning proper area coverage, considering agents' visibility, distance, or environmental disturbances. Four main topologies are implemented in this experiment: directed mesh with two lead agents (type A), directed mesh with one lead agent (type B), star topology (type C), and ring topology (type D). Type B and C topologies are well-suited for communication without delays or disturbances but perform poorly when the system experiences delays (e.g., noise disturbances exceeding a threshold frequency of 5 Hz). In contrast, type A and D topologies are more effective in handling communication delays. By implementing a topology-switching mechanism, this research streamlines the application of two or more topologies in real-life scenarios. It utilizes type B or C topologies in undisturbed conditions and switches to type A or D when perturbations occur. This optimization minimizes communication delays during mission execution and flight time. The research demonstrates significant improvements in trust evaluation, communication efficiency, and overall MAS performance, with implications across various domains, including image and video mining. In these areas, the integration of domain-specific agents enhances processes such as preprocessing, feature extraction, and interpretation. Results show improved accuracy and reliability in data analysis and decision-making across diverse applications, particularly in scenarios involving complex spatial objects and varying environmental conditions.

Keywords—Communication, Delay, Data Pooling, Delay, Optimization, Perturbation, Worldview

I. INTRODUCTION

The integration of a Multi-Agent System (MAS) across various stages of image and video mining—such as preprocessing, transformation, feature extraction, interpretation, and evaluation—offers significant potential to enhance both efficiency and effectiveness. By deploying domain-specific agents tailored to adaptable spatial objects, this approach addresses crucial factors like object visibility, agent-to-object distance, noise disturbance, and communication trust levels, likewise the trustworthiness of reported information. The objective is to build an advanced intelligent multi-agent system with minimal path planning algorithms, where agents utilize communication and initial task input to optimize self-navigation. The system focuses on extracting data, such as in image mining processes, resulting in improved accuracy and reliability of data analysis and decision-making across diverse applications. This approach differs from existing methodologies that focus on cooperative control of multi-robot systems (MRSs) for complex tasks. J. Hu et al. [1] emphasizes the flexibility, reconfigurability, and robustness of MRSs, whereas this study focuses on decision-making in image mining, particularly in areas such as data pooling data pooling, image clarity, and agent-to-object distance. J. Sumpter et al. [2] address communication delay and trust in MAS, analyzing system stability through graph theory and the Laplacian matrix. In contrast, this research employs physical tools like Laser Range Finders and Time-of-Flight (ToF) Cameras to achieve precise distance measurement during data collection. Additionally, N. Nguyen et al. [4] propose a decentralized Monte Carlo Tree Search algorithm to optimize action coordination in dynamic environments, whereas [2], [3], [10]-[12].

A. Method

In the area of data extraction, where multi-agent systems are being employed to accomplish certain tasks, decision-making relies on the precise evaluation of data pooled from multiple agents. However, certain underlying factors can affect the clarity of data, regardless of an agent's ability to capture the necessary information. These factors include distance, visibility (influenced by environmental conditions), perturbations, an agent's ability to operate in extreme conditions, and the effectiveness of ML/AI models. Developing a system that analyzes these parameters can enhance data extraction and evaluation during real-life missions. By incorporating these factors, agents can optimize

data collection and make informed decisions that enhance the overall efficiency and effectiveness of the system in achieving its objectives [13]-[15]. MAS is utilized in various applications, including manufacturing, transportation, healthcare, agriculture, geospatial analysis, and military operations. This work focuses on cooperative control, where only the initial departure point, distance to the area of interest, and the scope of the intended task are predefined. Agents are required to coordinate with each other to complete the required tasks without preprogrammed path planning or machine learning-based flight control. Instead, they rely on a specific communication pattern for coordination until the task is fully accomplished. Given these conditions, communication is pivotal, and any delay can pose significant challenges, slowing down the system and reducing efficiency. In some cases, such delays may even lead to system failure. To address this issue, star, mesh, and ring topologies were analyzed, and a switching mechanism was implemented as an optimized approach to minimize communication delays. This study also introduces a robust framework for improving MAS performance through innovative trust evaluation. Considering that communication and data sharing play a crucial role in MAS, it is essential to develop a model that evaluates trust in information exchange and ensures the authenticity of reported data. The trust model equation is applied in two scenarios:

Agent-to-agent trust evaluation – This assesses the trust coefficient based on whether an agent's position is safe or unsafe during task completion.

Agent-to-object position evaluation – This determines whether an idle position allows for efficient data retrieval, regardless of the agent's capability. Proximity to the area(s) of interest, whether too far or too close, may lead to inaccuracies in the information obtained. By addressing challenges such as communication delays, environmental disruptions, and data reliability, the proposed methods enhance the adaptability and effectiveness of MAS in real-world applications. These advancements pave the way for more efficient and secure multi-agent collaborations in domains requiring high-stakes decision-making and data pooling.

B. Data Pooling and Decision Making

The integration of data pooling and decision-making is critical for achieving cooperative control in Multi-Agent Systems (MAS). Data pooling involves aggregating and sharing information among agents, ensuring that all agents operate with a collective understanding of the environment and mission requirements. The data processing techniques depend on each agent's characteristics, including its hardware and processing capabilities. Agents apply image processing techniques, such as object detection, image segmentation, and feature extraction, to enable them to understand and interact with their environment. For example, agents can use object detection algorithms, like Convolutional Neural Networks (CNNs), to identify and track objects of interest within their visual field. Image segmentation techniques are used to divide the image into meaningful regions, allowing agents to focus on specific areas of interest for tasks like bin picking or obstacle avoidance. Feature extraction, including techniques like edge detection or texture analysis, helps agents recognize key attributes of objects, such as shape or

texture, enabling them to make more informed decisions and collaborate effectively with other agents. These image processing techniques allow agents to process visual data in real-time, enhancing their ability to perform complex tasks autonomously in dynamic environments. This process can be facilitated through shared information's among agents via a specific topology or shared memory models, where agents upload and retrieve data from a centralized or distributed memory system. Communication topologies, such as centralized (Star), decentralized (mesh), or dynamically switching networks (a system capable of dynamically switching between different topologies), also play a vital role in ensuring that information flows efficiently among agents. To further enhance data pooling by combining inputs from various sensors, such as cameras, LiDAR, and GPS, to create a comprehensive and accurate model of the environment.

Fig. 1 illustrates an automatic sensor switch designed for a multi-agent system application. The switch dynamically changes the system topology based on detected communication delays, often caused by high-frequency noise disturbances. When delays are identified, the switch transitions to a topology better suited for minimizing delays. For instance, while special case topology (A) and ring topology (D) are less effective under no-delay conditions compared to mesh and star topologies, they perform significantly better in managing delays. This adaptability makes the switching mechanism highly effective for dynamic environments. Distributed planning enables each agent to perform local computations for its actions while considering the global objective. For example, Dynamic Task Allocation Algorithms ensure that tasks are assigned based on an agent's proximity, capability, and workload. Additionally, consensus-based decision-making facilitates collaborative strategies through voting mechanisms or leader election, enabling agents to agree on shared objectives and prioritize tasks effectively. Before delving into the detailed step-by-step solution that involves calculating weighted scales for distance, visibility, trust, and then integrating these into determining the data pooling percentage based on the predicted accuracy, an overall equation that encapsulates the essence of the model can be considered [16]-[19]. The following parameters are carefully assigned to analyze or determine the trustworthiness of pooled (extracted) data from a specific agent. These parameters include: *distance* (D), *visibility* (V), *trust* (T), and predicted accuracy (A) to directly yield a decision-making outcome, such as the data pooling *percentage* (P).

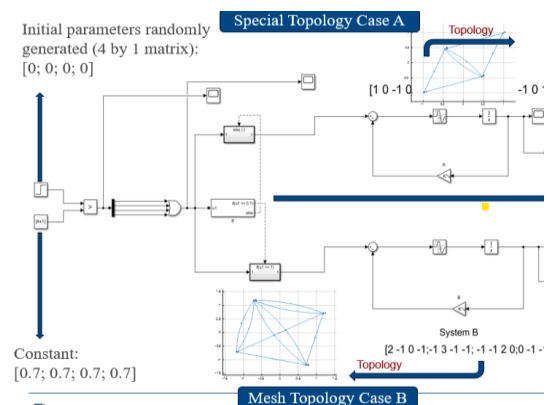


Fig. 1. Dynamic sensor switch for Case A and B topology. (automatic)

II. MATH

A. Overall Equation Concept

This overall equation presents a compact, integrative view of how the decision-making process is formulated based on the interplay of distance, visibility, trust, and agent accuracy within the MAS for image and video mining [20]. It is essential to recognize that multi-agent systems can operate in either a dynamic or fixed position. This mathematical model is applicable to both scenarios. However, in the current research environment, which focuses on a UAV fleet, the parameters continuously change due to variations in agents' flight or navigational formation. Since agents operate in a dynamic environment, the weighted parameters are not fixed [21]. Given the complexity and multi-faceted nature of the decision-making process described, an overall equation that accounts for all variables and their interactions could be conceptualized as follows [22], [23]:

$$P = f(w(D), w(V), w(T), A) \quad (1)$$

Where:

- P is the data pooling percentage, indicating how much weight or trust to allocate to data from a particular agent.
- $w(D), w(V),$ and $w(T)$ are the weighted scales for distance, visibility, and trust, respectively, each determined by attributed conditions at any given time.
- A is the predicted accuracy of the agent.
- f is a function that integrates these inputs to compute P , reflecting *the* decision-making criteria based on the model's requirements.

The function f can be designed to first calculate a composite weighted scale ($w(A)$) that amalgamates $w(D), w(V),$ and $w(T)$ with appropriate coefficients reflecting their importance [24], [25]. Then, it adjusts this composite scale based on A , applying different rules for different ranges of A :

$$w(A) = \alpha \cdot w(D) + \beta \cdot w(V) + \gamma \cdot w(T) \quad (2)$$

$$P = \begin{cases} \min(35\%, w(A) * 100\%), & \text{\mbox if } A < 35\% \\ 36\% + (w(A) * (75\% - 36\%)), & \text{\mbox if } 35\% \leq A < 76\% \\ 76\% + (w(A) * (100\% - 76\%)), & \text{\mbox if } A \geq 76\% \end{cases}$$

Given the requirements, let's construct a mathematical model to analyze and solve the problem described in the abstract, focusing on the integration of a Multi-Agent System (MAS) into image and video mining processes with an emphasis on decision-making based on agent-to-object distances, object visibility, and trust levels. The model involves three main variables which can directly affects agents' ability on data pooling or communication [26]-[28]:
Distance (D): Distance from the agent to the object, affecting visibility and decision-making.

Visibility (V): Quality of the object's visibility to the agent, influenced by environmental factors like lighting and obstructions.

Trust (T): Trust level associated with each agent, reflecting its security features and decision-making accuracy.

Let's denote:

- $w(D)$: Weighted scale based on distance.
- $w(V)$: Weighted scale based on visibility.
- $w(T)$: Weighted scale based on trust.

Important Variables: It is important to recognize that the weighting parameters assigned to the AI/ML capabilities and historical performance of the model do not have an immediate impact on an agent's ability at the time of data extraction [29]. However, these factors are given significant consideration when it comes to establishing trust and improving efficiency in communication and data pooling processes. Table 1 below highlights the parameters, detailing how they are measured, while Table 2 illustrates the representation of these parameters and their application in a mathematical format.

Table 1. Important Parameters

| |
|---|
| Data Pooling and communication analysis Communication Between Agent |
| Agent to agent = Distance |
| Agent to Agent = Perturbation |
| Agent to Agent = Trust Level |
| Agent = AI/ML |
| Data Pooling and communication analysis Data extraction By Agent |
| Agent to Object = Distance |
| Agent to Object = Visibility |
| Agent to object = Perturbation |
| Agent = AI/ML |
| Data Pooling and communication analysis Communication Agent to Mission Control |
| Agent History |
| Distance |
| Perturbation |
| Mission Control = AI/ML |

Table 2 outlines the key parameters that may impact an agent's ability to make informed decisions, retrieve accurate data, and report reliable information. These parameters may vary, as shown in Table 1.

Table 2. Variables Notation

| Distance(D) | Visibility(V) | Perturbation (Pb) | History | AI/ML |
|-------------|---------------|-------------------|---------|-------|
|-------------|---------------|-------------------|---------|-------|

B. Trust Equation

The trust equation and its results are significant for two key reasons: firstly, to ensure security, and secondly, to evaluate data accuracy, determining how reliable the reported information is. Trust between agents or from agent to mission control is one of the most important factors in this research [30]. Our approach on trust equation is important for several key reasons, especially in systems that require the assessment of reliability and credibility among entities, such as in multi-agent systems [31]. We implement trust equations to help with the following:

- **Quantitative Assessment:** Provides a quantitative measure of trust, allowing for the comparison and ranking of entities based on their trust scores.
- **Decision-Making:** The equation provides inform decision-making processes by highlighting the most reliable entities to engage with, collaborate with, or allocate resources to.
- **Consistency and Standardization:** Applies a standardized method to evaluate trust, ensuring consistency in how trustworthiness is assessed across different scenarios.
- **Risk Management:** By quantifying trust, agents can better manage risks associated with perturbations related

to hacking.

- **Incentivization:** Agents are incentivized to improve aspects such as performance, credibility, and accuracy, knowing these factors contribute to their trust score.
- **Predictive Analytics:** Trust scores can be used in predictive models to anticipate future behaviors based on past performance and reliability.
- **Security and Fraud Detection:** Higher trust scores can indicate lower risk, which is crucial for security and fraud detection mechanisms.

Below is the formula for the Trust score equation [32], [33]:

$$T = \alpha \cdot P + \beta \cdot C + \gamma \cdot A \quad (3)$$

Analysis of Distance, Visibility and Trust Between Agents

Decision-making: Since environmental conditions can influence an agent's ability to make informed decisions and retrieve reliable data. Agents apply equation 3, as an evaluation metric on nearby agents to determine a trustworthiness score, estimating the accuracy percentage of communicated data received. Based on this assessment, they adjust navigational instructions according to the communication topology pattern established during the operation [34]. If an agent's evaluation metrics for a nearby agent fall below the 50% threshold, it initiates communication instructions based on the applied topology, directing the underperforming agent to adjust its position for more accurate data retrieval.

Table 3 above highlights the measurement of each parameter for every agent and explains how the overall score of the agents is evaluated during a relief scenario. There is an appropriated distance per position ratio denoted for adequate or precise information retrieval.

Table 3. Trust level weighted parameters (refer to table attached file)

Kindly refer to table sheet

C. Distance Weighted Scale

When considering the dynamic behaviors of agents within a multi-agent system, particularly regarding their positional analysis, evaluating the consistent variation in distance over an unbounded temporal scope can be critical. Within a natural setting, one can demarcate a region or range as safe zone, while areas beyond this boundary may be considered as unsafe zone. Such regions are where agents are more vulnerable to risks. When it comes to pinpointing the location of agents that traverse at constant rates or experience unchanging acceleration, the principles of kinematics become highly relevant [35]. The determination of which kinematic equation to use hinges on the specific characteristics of the movement—whether it involves consistent speed, uniform acceleration, or the impact of other dynamic factors [36]. To determine the distance between agents/mission control, or from the object of interest to the Agent. Agents are either equipped with sensor, GPS, or sensors to apply the following formulas for fixed and Dynamic motion. For a fix Object and with agent in a fixed position the following formula is denoted for agent to be able to analyze its distance [37]: To calculate the distance between two fixed points, you can use several different methods depending on the context and the type of data you have for these points. Here are two common

scenarios and the corresponding methods for calculating distance:

For Points in a Cartesian Plane (2D or 3D):

- **Two-Dimensional Space (2D):** If you have two points, $A(x_1y_1)$ and $B(x_2y_2)$, the distance between them in a 2D Cartesian plane is given by the Euclidean distance formula:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (4)$$

- **Three-Dimensional Space (3D):**

For points $A(x_1y_1z_1)$ and $B(x_2y_2z_2)$, in 3D space, the distance is calculated as:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2} \quad (5)$$

For Constant Velocity of Agent: If the change in distance is due to constant velocity (which implies no acceleration), the formula is straightforward:

$$d = v * t \quad (6)$$

Where:

- d is the distance traveled,
- v is the constant velocity
- t is the time elapsed.

This formula implies a linear relationship between distance and time.

For Constant Acceleration: Considering a situation involving constant acceleration, such as free-fall under gravity, the formula from kinematics that describes the distance traveled over time is:

$$d = v_0t + \frac{1}{2}at^2 \quad (7)$$

Where:

- d is the distance,
- v_0 is the initial velocity,
- a is the constant acceleration,
- t is the time.

The examination of an agent's distance and position is pivotal as it can significantly impact the agent's ability to communicate with its peers, as well as potentially influence its interaction with or perception of a targeted object or an assigned mission. Accurate assessment of these spatial parameters is essential to ensure optimal functionality within the system, facilitate seamless coordination among agents, and enhance the successful completion of tasks. Misjudgments in positioning may lead to compromised communication and inefficiencies in task execution, thereby underscoring the importance of precise location analysis in maintaining the efficacy and safety of multi-agent operations.

D. Weighted Scale-based on Visibility/Object Recognition

When assessing visibility affected by natural perturbations such as weather conditions or the blurriness of an object, the best optical equations to analyze the situation typically derive from the Modulation Transfer Function (MTF). MTF provides a measure of the optical system's ability to transfer contrast at a particular resolution from the object to the image. The various criteria is scrutinized and considered during data pooling for decision-making. If an agent's training model achieves a high accuracy of 95 percent,

then any predicted result related to the weighted category will be assigned accordingly [37]:

The MTF is affected by various factors, such as diffraction, aberrations, and defocus—all of which can contribute to blurriness. In atmospheric conditions, scattering due to particles or variations in refractive index (e.g., due to rain, fog, or heat haze) can also impact visibility [38].

Basic MTF Equation: The basic MTF of an optical system can be described as:

$$MTF = \left| \frac{\text{Image Contrast at Spatial Frequency } f}{\text{Object Contrast at Spatial Frequency } f} \right| \quad (8)$$

Where:

$MTF(f)$ is the modulation transfer function at spatial frequency f ,

The image contrast is the contrast in the image produced by the optical system,

- The object contrast is the actual contrast present in the object itself,
- The spatial frequency f refers to the level of detail in the object, typically measured in cycles or line pairs per millimeter.

Visibility in Weather Conditions: For visibility affected by weather, you could incorporate the contrast attenuation due to scattering and absorption, which is described by the Koschmieder's Law:

$$V = \frac{1}{k \cdot \beta} \quad (9)$$

Where:

- V is the visibility in kilometers.
- k is the extinction coefficient, which depends on factors like aerosol concentration, air pollution, and weather conditions.
- β is the luminance contrast between the object and its background.

Blurriness due to Optical Aberrations: Blurriness caused by optical aberrations, including weather-induced perturbations like scattering, can also be modeled by incorporating the Point Spread Function (PSF) into the MTF, as MTF is the Fourier Transform of the PSF.

Complete MTF Model for Visibility: A complete MTF model that accounts for visibility under natural perturbations could look like:

$$MTF_{total}(f, B) = MTF_{optical}(f) \cdot e^{-B \cdot V(f)} \quad (10)$$

Where:

- $MTF_{optical}(f)$ is the MTF due to the optical system itself,
- $e^{-B \cdot V(f)}$ is the attenuation factor due to weather conditions, with $V(f)$ being a function describing how visibility changes with spatial frequency, potentially modeled from experimental data or additional atmospheric physics equations.

For a computational model or simulation, you'd combine these optical equations with empirical data about the specific visibility conditions you're studying to analyze how various factors contribute to overall visibility and the ability to recognize objects. This model can guide decision-making about the reliability of visual information under various environmental conditions.

1) Perturbation Analysis

Effectively addressing perturbations such as Light pollution, high wind, and noise is crucial for the resilience and reliability of a MAS, particularly in complex and variable environments where external disturbances are likely. Each of these perturbations requires specific strategies for mitigation [39]. **Noise (high wind or light pollution):** Sensors to measure high winds and other perturbation or other Natural disturbances such as light reflection [40], [41]. To Measure Light reflection, Calculate the distance from Object to Agent. From the Agent **latitudes and longitude position and the distance**, Determine the angle at the time an image was taken.

➤ Obtain real time weather conditions including the time at which an image or video was taken to analysis any light disturbance.

➤ latitudes and longitudes of two points on the Earth's surface in radians, and R is the Earth's.

Fig. 2 shows the different level or altitude of RVR – Runway visual range, SVR – Slant Visual range Despite its variability, the sole fog parameter currently measurable and reportable is the (RVR), which gauges atmospheric opacity at a low height. Fig. 2 shows an example of visual range with agents on different positions (range). However, RVR does not consider vertical density changes, and in cases of increasing density with height, any SVR will be shorter than the RVR.

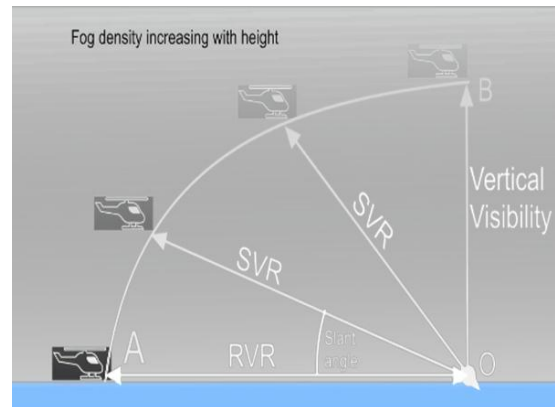


Fig. 2. Visual range

2) Image Processing and Object Recognition

In computer vision and artificial intelligence, the general concept of a mathematical model that could be used for object recognition [42]. Object recognition involves identifying specific objects within an image or a video. The process usually involves several steps [43]:

Preprocessing: This includes noise reduction, normalization, and other techniques to prepare the image data for further analysis.

Feature Extraction: Key features of the objects are extracted to be used for recognition. This could involve edge detection, texture analysis, color histograms, or more sophisticated features like SIFT (Scale-Invariant Feature Transform) and SURF (Speeded Up Robust Features).

Classification/Recognition: The extracted features are then fed into a classification model that has been trained to recognize various classes of objects. This model could be based on machine learning algorithms such as support vector machines (SVM), neural networks, deep learning models like convolutional neural networks (CNNs), or other statistical

models [44]. A simplified mathematical model that encapsulates this process can be broken down as follows:

Preprocessing Function P:

$$X' = P(X) \quad (11)$$

- X is the original image data.
- X' is the preprocessed image data.

Feature Extraction Function F:

$$\text{features} = F(X') \quad (12)$$

features is the vector or set of extracted features from X' .

Classification Function C:

$$\text{Label} = C(\text{features}; \theta) \quad (13)$$

- label is the recognized class of the object.
- θ represents the parameters of the classification model, which are learned during a training process from a set of labeled

For example, in the context of a multi-agent system, this recognition process might be distributed among various agents. For example, different agents could handle different parts of the image or specialize in different aspects of the recognition process. The collective outputs would then be integrated to arrive at a final decision [44], [45]. The overarching model for a distributed object recognition system in a multi-agent framework could be expressed as a fusion of the outputs from individual agents:

Integration Function I:

$$\text{object} = I(\text{label}_1, \text{label}_2, \dots, \text{label}_n) \quad (14)$$

- object is the final recognized object after considering the inputs from all agents.
- label_i represents the output from the i^{th} agent.

The model would involve a strategy for how these individual recognitions are combined, whether by a voting mechanism, a probabilistic model, or some form of hierarchical decision-making. The goal is to leverage the strengths of each agent to improve overall accuracy and reliability in object recognition tasks.

3) Collaborative Image Extraction and Data Pooling

Using a Multi-Agent System (MAS) for object recognition entails distributing the object recognition tasks across multiple agents. Each agent can either specialize in a particular aspect of recognition or work on recognizing objects within different segments of the data. Here's a broad outline of how MAS can be implemented for object recognition, along with a mathematical model framework [46]:

Distributed Feature Extraction: In a MAS, feature extraction can be distributed among various agents. Each agent may handle different features, like color, shape, or texture [47]. Let agent A_i be responsible for extracting feature set F_i from image X . The feature extraction by each agent can be denoted as $F_i = E_i(X)$, where E_i is the extraction function.

Collaborative Classification: Agents collaborate to classify objects based on the combined features extracted:

Each agent classifies a part of the object or contributes a vote/classification decision based on its feature set. The classification by each agent can be denoted as $C_i(F_i)$, resulting in a label L_i .

Consensus and Integration: The agents need to reach a consensus or combine their findings to make a final classification decision: Integration of the agents' classifications can be modeled as a decision fusion process $D(L_1, L_2, \dots, L_n)$, resulting in the final decision L .

Feature Extraction:

$$\forall i \in \text{Agents}, \quad F_i = E_i(X)$$

Individual Classification:

$$\forall i \in \text{Agents}, \quad L_i = C_i(F_i)$$

Consensus/Integration:

$$L = D(L_1, L_2, \dots, L_n),$$

Here, D could be a simple majority voting, weighted decision-making based on the confidence of each agent, or a more complex model such as Bayesian integration or Dempster-Shafer theory [48].

Simulated Experiment: With the application of our three agents in the MAS: A_1 for color features, A_2 for texture features, and A_3 for shape features. Each agent

Extracts its respective features from the image and then performs an individual classification:

- A_1 examines the color histogram of the image to identify possible object matches based on color profiles.
- A_2 analyzes the texture using filters or other texture analysis methods to categorize the object.
- A_3 focuses on the outline and contour patterns to determine the shape category of the object.

After each agent has made a classification decision based on its feature set, a central system or another agent in the MAS tasked with decision integration takes the input from all agents $L = D(C1(F1), C2(F2), C3(F3))$.

Function D may involve applying weights to each agent's classification depending on their reliability and track record or may use machine learning models to learn the best way to combine inputs for accurate object recognition. The MAS approach to object recognition has several advantages, including parallel processing, which can reduce computation time, and increased robustness, as the system does not rely on a single point of failure and can handle diverse and complex environments. This is particularly useful in practical scenarios where different types of information need to be synthesized to accurately recognize objects in various conditions and contexts.

E. Agents Performance History

The determination of drone performance over time will be based on flight history, tracking key metrics such as the number of tasks or missions successfully completed versus those attempted within a given period. This includes verifying the drone's adherence to safety and operational standards, as well as monitoring operational parameters like flight time, battery life, payload capacity utilization, and distance covered. Additionally, instances of malfunctions, required maintenance, or mission failures will be recorded to assess overall reliability and performance.

F. Data Pooling

Data pooling in a Multi-Agent System (MAS) significantly enhances system resilience by ensuring robustness and fault tolerance through multiple data pathways, allowing for self-healing and rerouting if

individual agent fails [49]. This decentralized approach offers scalability, facilitating the seamless integration of additional agents, and ensures that the network's functionality isn't compromised by the failure of any single agent [50]. Considering a specific topology such as the Mesh topology, the inherent flexibility of mesh networks supports dynamic adaptability, crucial for managing complex systems and varied communication needs, and enhances overall communication reliability, making it an invaluable framework for applications that demand high resilience and adaptive capabilities.

The mesh topology graph depicted above in Fig. 3, illustrates the pattern of communication among the agents. Referencing Equation 1, the diagonal entries of Table 4, which is the weighted scale matrix, represent the predicted accuracy "A" for each respective agent. This reflects the absence of self-communication; that is, an agent does not communicate with itself [51]. Therefore, the accuracy "A" is computed as a weighted sum that incorporates various attributes and the results of an analytical assessment, all in relation to the specific parameters outlined in the Agent-Self-Evaluation displayed in Table 4. These parameters are critical in determining the overall effectiveness and reliability of each agent within the network, considering the agent's performance based on internal metrics and past interactions as documented in the detailed self-assessment provided [52].

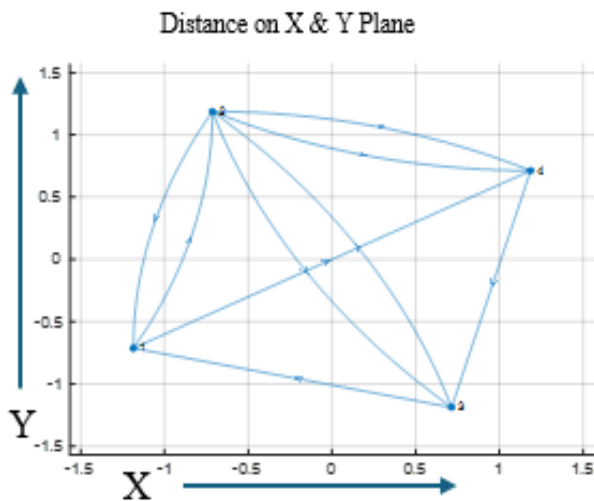


Fig. 3. Graph of a mesh topology

Table 4. Weighted scale Matrix (4 by 4)

| Agents | A_1 | A_2 | A_3 | A_4 |
|--------|------|------|------|------|
| A_1 | 0.7 | -0.4 | 0 | -0.7 |
| A_2 | -0.8 | 0.9 | -0.8 | -0.8 |
| A_3 | -0.7 | -0.4 | 0.7 | -0 |
| A_4 | 0 | -0.5 | -0.4 | 0.8 |

$$P = f(w(D), w(V), w(T), A)$$

Table 5 shows how mission control or every agent evaluates itself and provides an initial reported score prior to task accomplishment. It should be noted that the predicted accuracy 'A' per Agent can also be calculated using equation 2.

Table 5. Agent-Self Evaluation Table

| Quantitative Measurements | Agent to Agent | Agent Score |
|-------------------------------------|-------------------------------|-------------|
| Communication Performance Score (P) | Communication History | 0.05 |
| | Operational Metrics | 0.05 |
| | Data Analysis | 0.2 |
| Delay & Trust Credibility Score (C) | Agent to Agent distance | 0.1 |
| | Safe Zone/ Visibility | 0.1 |
| | Reputation score Perturbation | 0.1 |
| | Navigation/ (AI/ML) | 0.1 |
| Data Pooling Accuracy Score (A) | Sensor & Data Accuracy | 0.1 |
| | Task Execution | 0.05 |
| | Quantitative Analysis | 0.15 |

1) Data Pooling and communication between Agents

Effective communication in MAS is typically characterized by its robustness, reliability, and adaptability, allowing the system to function harmoniously in dynamic environments [53]. The communication protocols must be designed to handle potential disruptions and to optimize the flow of information, ensuring that each agent functions as part of an integrated whole, despite having autonomous capabilities [54]. This interconnectedness is crucial for complex tasks such as distributed problem-solving, synchronized actions, and collective learning within the MAS framework. When analyzing data pooling and delays in communication between agents, it is imperative to approach this issue with a weighted scale that allows agents to evaluate one another before establishing trusted communication. The level of trust gained by an agent could also be analyzed by which agent has the most ability to communicate or assign tasks. The weighted scale concerning communication and data pooling differs across interactions, whether between agents or between an agent and mission control. This variance is contingent upon the significance attributed to the five parameters mentioned earlier: distance, visibility, perturbation, AI/ML capabilities, and historical data. It's important to note that each agent may possess distinct characteristics or rankings. For instance, unlike mission control, certain agents may be equipped with highly advanced sensors, computational power, and machine learning capabilities.

2) Data Pooling and Communication Between Agent and Mission Control

Data pooling and communication between agents and mission control are critical components in the operation of Multi-Agent Systems (MAS), especially in contexts where coordination, decision-making, and efficiency are paramount. This process involves aggregation, analysis, and exchange of information collected by individual agents to a centralized mission control or among the agents themselves [55]. Here's an overview of how these mechanisms work and their importance, **Communication:**

Bilateral Exchange: Communication between agents and mission control is typically bidirectional, allowing for both the transmission of collected data to mission control and the reception of commands or updates from mission control to the agents.

Protocols and Standards: Effective communication relies on established protocols and standards to ensure that data is exchanged in a secure, reliable, and interpretable manner.

Real-time Processing: For many applications, the ability to communicate and process data in real-time or near-real-time is crucial, especially for time-sensitive operations.

Adaptability and Scalability: Communication mechanisms must be adaptable to changes in the operational environment and scalable to accommodate an increasing number of agents or more complex tasks.

Given that mission control boasts superior computational capabilities, along with more efficient and advanced software for data analysis, the criteria used for determining the weighted scale significantly differ from that of other components. Table 6 above outlines the specific criteria utilized to define each category within the weighted scale. Consequently, when it comes to the trust aspect of the weighted scale, a comparable methodology is employed to populate the 4x4 matrix outlined below in Table 7. This process is crucial for establishing trusted communication among agents prior to the final sharing of data with mission control [56]. This approach ensures that the trustworthiness of communications is rigorously assessed and verified, leveraging mission control's analytical advantages to enhance the overall integrity and reliability of data exchange within the system.

Table 6. Criteria for weighted coefficient between agent and mission control

Kindly refer to table sheet

Table 7. 4x4 matrix

| Agent | A_1 | A_2 | A_3 | A_4 |
|-------|------|------|------|------|
| A_1 | 0.7 | -0.4 | 0 | -0.7 |
| A_2 | -0.8 | 0.9 | -0.8 | -0.8 |
| A_3 | -0.7 | -0.4 | 0.8 | -0 |
| A_4 | 0 | -0.5 | -0.4 | 0.8 |

Data pooling and communication between agents and mission control form the backbone of effective MAS operations. They not only enhance the system's capacity to perform its intended functions but also enable a level of adaptability and resilience in dynamic or uncertain environments. As MAS continues to evolve and find new applications, the mechanisms for data pooling and communication will remain central to their success and effectiveness.

G. Evaluation of Pooling Result

When it comes to decision-making in data pooling, thorough evaluation of various criteria is a critical process. During this assessment phase, the focus is on determining the relevance and reliability of data contributed by each agent. The underlying principle is to establish a weighted system that reflects the quality of data based on several key factors, which include the agent's predictive accuracy, operational environment, and trustworthiness [57]. For instance, if an agent's algorithm or model is rigorously trained and validated to achieve a high predictive accuracy—say, 95 percent—this high level of accuracy influences the weight assigned to the data it provides. In the context of the weighted categories provided, results predicted by this agent would be highly valued and likely receive a larger share of influence in the overall data pooling process. This is depicted in the matrix where the high accuracy falls within the top tier of the

weighted scale, leading to a greater data pooling percentage ranging from 86 to 100 percent.

This tiered system ensures that data from more accurate and trusted sources has a proportionately greater impact on collective decisions. As such, an agent operating with high predictive accuracy and in a safe zone with better visibility is considered dependable, and its contributions are weighted accordingly to reflect its higher reliability. Conversely, data from agents with lower predicted accuracy or from less trusted zones would contribute less to decision-making, as denoted by the lower data pooling percentages associated with these categories. The stratified approach, as laid out in the accompanying matrix in Table 8, aligns the decision-making process with the overall integrity and reliability of the pooled data, ultimately guiding the MAS towards more informed and reliable outcomes [58].

Table 8. Decision making Process.

| | | |
|---|---|---|
| Poor visibility, further away, From untrusted zone, low predicted accuracy (Poor) | Image predicted accuracy = 50 to 85%, within safe zone of operation, better visibility. | Image predicted accuracy = 86 to 100%, within safe zone of operation, Dependable (Trusted). |
| Data Pooling Percentage = 0 to 49% | Data Pooling Percentage = 50 to 85% | Data Pooling Percentage = 86 to 100% |

1) Discrete analysis of Data Pooling

Research on data pooling for worldview coordination in contested communication spaces, especially within a multi-agent system (MAS), is increasingly pertinent. The MAS paradigm facilitates a dynamic collaboration environment where multiple autonomous agents gather, process, and exchange data to arrive at consensus decisions. By distributing tasks and allowing agents to dynamically interact, MAS can refine the identification process and escalate decision-making accuracy [59], [60]. This distributed intelligence enables robust and adaptive solutions for a myriad of applications, from resource management to complex routing and autonomous operational systems.

The decision-making process within this framework is comprehensive. It involves critical assessment of data quality, where factors such as clarity and potential distortions like blurriness are meticulously evaluated. Moreover, assessing the spatial dynamics between agents and target objects can further inform decision-making, drawing upon weighted considerations during the data collection phase.

A consolidated equation that succinctly embodies the essence of the decision-making model would include weighted scales for distance (D), visibility (V), trust (T), and predicted accuracy (A) of each agent: $P = a \cdot w(D) + b \cdot w(V) + c \cdot w(T) + d \cdot w(A)$

This equation signifies the data pooling percentage (P) that integrates the mentioned variables, where 'f' represents the function that binds these inputs together, reflecting the decision-making protocol tailored to the system's requirements.

To explicate the function 'f,' one could articulate a composite weighted scale ($w(A)$) that blends $w(D)$, $w(V)$, and $w(T)$ using tailored coefficients indicative of their relevance. Subsequently, this amalgamated scale is adjusted based on A, enacting diverse criteria contingent upon A's

range, referring to equation 2 above. In the model underpinning our Multi-Agent System (MAS), the data pooling percentage is determined by a dynamic scale which changes in response to the predicted accuracy (A) of an agent's outputs. The structure of this scale is as follows:

- If an agent's predicted accuracy is less than 49%, the data pooling percentage is capped at the minimum of 49% or the weighted scale ($w(A)$) multiplied by 100.
- For a predicted accuracy between 50% and 85%, the data pooling percentage is set at a base of 50% plus an additional percentage derived from multiplying the weighted scale ($w(A)$) by the difference between 85% and 50%.

When the predicted accuracy is 86% or higher, the percentage begins at a base of 86% plus an incremental percentage, which is the product of the weighted scale ($w(A)$) and the difference between 100% and 86%. The evaluation criteria focusing on distance, visibility, and trust significantly enhance the multi-agent system's (MAS) capability to process and interpret image and video data with high integrity. The system maintains a comprehensive performance history for each agent, detailing task success rates and adherence to operational metrics, which are essential for establishing credibility and predicting reliability. Additionally, the MAS demonstrates resilience against various disruptions, such as noise, high winds, and cybersecurity threats, thanks to its robust design and the agents' adaptive navigation abilities. The trust equation, which underpins communication among agents and between agents and mission control, highlights the importance of quantitative measures in assessing credibility, thereby ensuring effective decision-making within the MAS. This framework motivates agents to continuously improve their performance, enhance security, and leverage predictive analytics for a robust and responsive system.

Experimental setup and evaluation: The experiment was conducted in a controlled outdoor environment designed to simulate various applied conditions, using four commercially available drones equipped with high-resolution cameras and relevant sensors. The drones were programmed to follow predefined flight paths while capturing images and data under varying environmental conditions, including different light levels, wind speeds, and temperature ranges. Key performance metrics such as image quality, data capture accuracy, flight stability, and maneuverability were recorded and analyzed. The evaluation focused on assessing the drones' ability to capture detailed and accurate data, adapt to changing conditions, and maintain consistent data capture rates. The results provided valuable insights into each drone's performance characteristics and their suitability for specific data extraction tasks across diverse scenarios.

The weather conditions depicted in Fig. 4, corresponding to the time and day of the experiment, proved to be vital and immensely beneficial for assessing the agents' performance under various weather scenarios. Each agent was equipped with its own machine learning algorithm, meticulously trained across seven distinct classes of data. These classes encompassed a diverse range of visual patterns and features, enabling the agents to develop a comprehensive understanding of various objects and scenarios. Through extensive training, each algorithm was adept at recognizing

and categorizing objects within its designated class, ensuring a robust foundation for subsequent tasks.



| Day (1-5) | Local Time | Direction | Speed/Gust | Temp/Feels Like | Humidity | Pressure | Cloud Amount | Precip Amount | Weather |
|-----------|------------|-----------|-------------|-----------------|----------|----------|--------------|---------------|---|
| 4/12/24 | 16:00 | WNN | 17 / 19 mph | 63 / 62°F | 45% | 1008 mb | 11% | 0.00 In |  |
| 4/11/24 | 13:00 | W | 19 / 22 mph | 64 / 63°F | 38% | 1008 mb | 30% | 0.00 In |  |

Fig. 4. Weather conditions during experiments

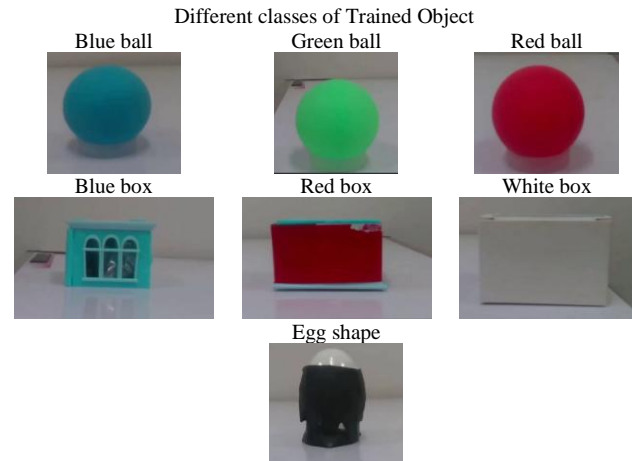


Fig. 5. Different classes of trained object

This meticulous training process equipped the agents with the knowledge and capability to accurately identify objects encountered during their operations, enhancing their overall effectiveness and performance. Fig. 5 show the different types and shapes of object. Every class was equipped with a training dataset containing approximately 400 to 700 images. This diverse set of images ensured that the machine learning models had ample data to learn from, enhancing their ability to recognize and classify objects accurately. Following the execution of the machine learning algorithm, agents were capturing a single object. Each agent, leveraging its respective training model, was tasked with identifying the captured object.

The deployment involved a well-coordinated approach where agents worked in concert, employing their unique perspectives and vantage points to gather comprehensive data about the object. Each agent's training model, fine-tuned with the robust dataset, enabled it to independently identify and analyze the object from different angles and under various conditions. This collective effort not only improved the accuracy of object identification but also provided a more holistic understanding of the object's features. Fig. 6 provides insights into the machine learning performance metrics for Agent 4, including its confusion matrix, which shows classification accuracy, and the training accuracy, reflecting the model's overall performance. Each agent in the multi-agent system operated with its own machine learning model, leading to variations in accuracy scores across agents. This tailored approach optimized each model based on the agent's operational requirements and environmental constraints. The confusion matrix and training accuracy for Agent 4 offer a clear understanding of its model's classification capabilities and effectiveness in fulfilling its tasks.

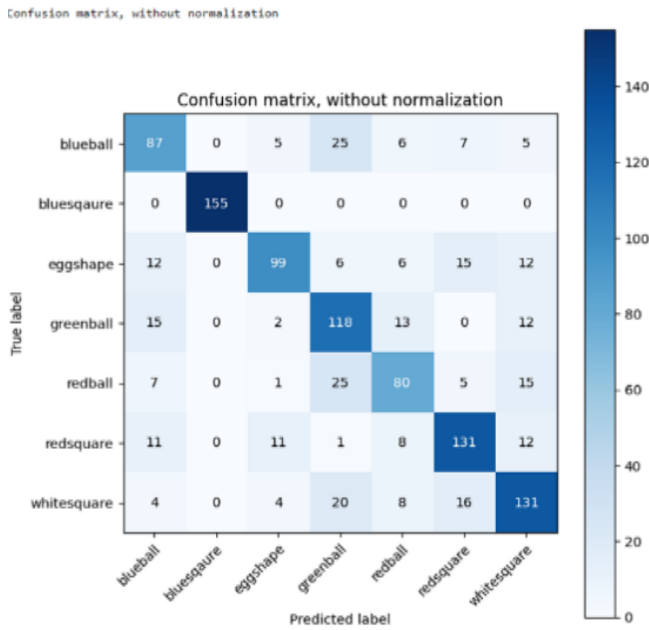


Fig. 6. Agent 4 machine learning - confusion matrix

Examining Fig. 7 elucidates the discernible absence of agent 4 from the operational milieu, delineating its occupancy of what may be deemed an insecure domain. Conversely, agent 3 finds itself ensconced within the confines of the secure zone, albeit distanced from the target entity. Within the precincts of safety also reside agents 1 and 2, yet agent 1 finds itself perilously proximate to the target, a condition deemed less than optimal. Conversely, agent 2, ensconced in a felicitous locale, exemplifies an ideal position within the operational context.



Fig. 7. Map out of operational environment experiment 1

Table 9 provides a comprehensive matrix representation illustrating the distances, measured in meters, between each pair of agents within the system. This matrix format offers a structured overview of the spatial relationships and proximity between the various agents, facilitating an understanding of their relative positions and interactions within the operational environment. On the other hand, Table 10 offers specific details regarding the distance, also measured in meters, from each individual agent to the designated target entity. By delineating the distance metrics for each agent separately, this table offers a focused perspective on the spatial dynamics and proximity of each agent to the target object. Such detailed insights are essential for assessing the operational effectiveness and strategic positioning of the agents within the system, particularly in relation to their objectives and operational constraints.

From the above weather conditions shown in Fig. 4 corresponding to day 1. For atmospheric conditions, the

extinction coefficient k tends to be lower, indicating less scattering and absorption of light. It should be noted that typical value for k under clear conditions might be around 0.1 per kilometer, but this can be valued based on factors such as air pollution and aerosol concentration. Meanwhile the estimation of Luminance contrast (β) depends on the specific environment and the observer's perception of contrast between the object and its background. With a clear background around the object, it was assuming a moderate contrast level, which translated to a value of $\beta=0.5$. Using $k = 0.1$ per km and $\beta = 0.5$, we get:

Table 9. Distance between agent in matrix representation

| Distance (m) b/w Agents | Agent_1 | Agent_2 | Agent_3 | Agent_4 |
|-------------------------|---------|---------|---------|---------|
| Agent_1 | 0 | 4 | 1.5 | 7 |
| Agent_2 | 4 | 0 | 5 | 2.5 |
| Agent_3 | 1.5 | 5 | 0 | 8 |
| Agent_4 | 7 | 2.5 | 8 | 0 |

Table 10. Distance from agent to object

| Agent to Object Distance | Object distance in meters(m) |
|--------------------------|------------------------------|
| Agent_1 | 1 m: away target |
| Agent_2 | 2 m: away target |
| Agent_3 | 3 m: away target |
| Agent_4 | 5 m: away target |

$$V = \frac{1}{k \cdot \beta} = \frac{1}{0.1 \times 0.5} = 20 \text{ km}$$

So, the visibility of agent 1 is---- approximately 20 kilometers.

Applying the same visibility due to weather conditions on the following agents as demonstrated for agent_1 above, the respective value for the following agents is shown on Table 12.

Experiment 1: Four Agent

Table 11, Table 13, and Table 14 present the visibility scores, accuracy scores, and trusted pooling evaluation scores for each agent under actual operating conditions. These scores are used to assess retrieved data and can also be applied in agent-to-agent communication for enhanced decision-making.

Table 11. Agent trust score evaluation

Kindly refer to table sheet

Table 12. Agent visibility due to weather conditions

| Agent visibility due to weather conditions | | | | |
|--|-----------------|--------------|-----------------|----------------|
| weight | Agent_1 | Agent_2 | Agent_3 | Agent_4 |
| k | 0.1 | 0.2 | 0.3 | 0.5 |
| β | 0.5 | 0.5 | 0.5 | 0.5 |
| V | 20,000 m = 0.50 | 10,000 m = 1 | 6666.7 m = 0.70 | 4,000 m = 0.40 |

- Solving for the predicted accuracy (A) of each agent.

$$w(A) = \alpha \cdot w(D) + \beta \cdot w(V) + \gamma \cdot w(T)$$

Table 13. Agents Predicted accuracy

| Agent Predicted accuracy (A) | | | | | |
|------------------------------|------------|-------------|-------------|------------|------------|
| Weight | Agent1 | Agent2 | Agent3 | Agent4 | Parameter |
| $\alpha = 0.3$ | 0.5 | 1 | 0.7 | 0.4 | Distance |
| $\beta = 0.3$ | 0.5 | 1 | 0.7 | 0.4 | Visibility |
| $\gamma = 0.4$ | 0.7465 | 0.9425 | 0.895 | 0.650 | Trust |
| A | 0.6 | 0.97 | 0.78 | 0.5 | |

$$P = a \cdot w(D) + b \cdot w(V) + c \cdot w(T) + d \cdot w(A)$$

Given that: $a + b + c + d = 1$, we can choose appropriate weights for the parameters. Let's set: $a=0.2$, $b=0.2$, $c=0.4$, $d=0.2$

Table 14. Credibility Pooling Percentage For every Agent

| Agents Pooling accuracy (P) | | | | | |
|-----------------------------|--------------|--------------|--------------|--------------|---------------------|
| Weight | Agent 1 | Agent 2 | Agent 3 | Agent 4 | Parameter |
| $a = 0.2$ | 0.5 | 1 | 0.7 | 0.4 | Distance |
| $b = 0.2$ | 0.5 | 1 | 0.7 | 0.4 | Visibility |
| $c = 0.3$ | 0.7465 | 0.9425 | 0.895 | 0.650 | Trust |
| $d = 0.3$ | 0.6 | 0.97 | 0.778 | 0.50 | Predictive accuracy |
| P | 0.604 | 0.974 | 0.783 | 0.505 | |

From the above credibility pooling percentage, a concrete analysis on understanding the validity of observed object could be determined.

Analyzing the pooling percentage of each agent, as depicted in Fig. 8, reveals compelling insights into their individual performances. Notably, Agent 2 emerges as the standout performer, showcasing remarkable levels of performance, credibility, and accuracy with an impressive percentage of 97.4. Such a high percentage underscores the reliability and effectiveness of Agent 2 in the data pooling process. This finding instills confidence in the accuracy of the identified object, depicted as a blue square in Fig. 6. It validates the selection of this object for experimentation, affirming its relevance and significance in the context of the study.

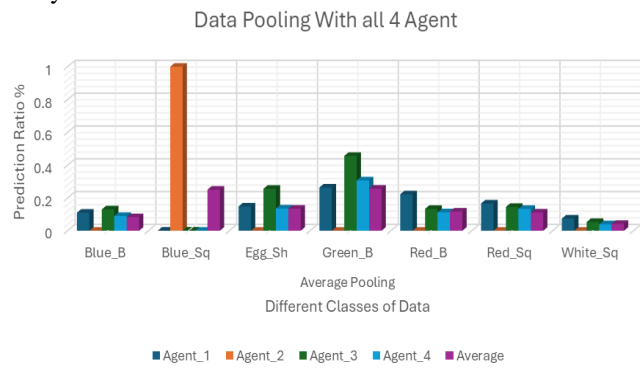


Fig. 8. Applying machine learning during data pooling

Experiment 2: Four Agent

A second experiment was conducted with slight changes in Agents positions, which affected the previous distance in experiment 1. Parameters for weather conditions were used in respect to day 2 as shown on Fig. 9.

Examining Fig. 9 elucidates the discernible absence of agent 4 from the operational milieu, delineating its occupancy of what may be deemed an insecure domain. Conversely,

agent 3 finds itself ensconced within the confines of the secure zone, albeit distanced from the target entity. Within the precincts of safety also reside agents 1 and 2, yet agent 1 finds itself perilously proximate to the target, a condition deemed less than optimal. Conversely, agent 2, ensconced in a felicitous locale, exemplifies an ideal position within the operational context.



Fig. 9. Map out of operational environment experiment 2

In the experimental setup for experiment 2, Table 14 provides the distances between agents in matrix format, while Table 15, illustrates the distances between the object and each agent.

A similar strategy to that demonstrated in Experiment 1 for evaluating trust among agents was implemented in the second experiment. Table 16 illustrates the evaluation scores for trust levels among agents in the second experiment. Utilizing the weather conditions specified on day 2 in Fig. 9, assumptions were made, and calculations of agents' visibility were conducted. The extinction coefficient (k) was estimated to be 0.1 per kilometer, like that of Experiment 1. Likewise, the estimation of luminance contrast (β) was set at 0.5. With a clear background surrounding the object, a moderate contrast level was assumed, resulting in a value of $\beta=0.5$.

The results from Table 14 (Experiment 1) and Table 18 (Experiment 2) are evaluated as follows: The implemented mathematical model provides a reliable framework for monitoring and determining the percentage of data that can be deemed trustworthy for accurate decision-making. The percentage differences in agent reading, particularly in pooling accuracy, reflect the trustworthiness of agents in reporting reliable information. This evaluation is especially critical for geospatial analysis and healthcare applications, where data integrity is essential. Agents Trust Score Evaluation- For Experiment 2 shown in Table 17.

Table 15. Distance between agents in matrix representation

| Distance (m) b/w Agents | Agent_1 | Agent_2 | Agent_3 | Agent_4 |
|-------------------------|---------|---------|---------|---------|
| Agent_1 | 0 | 4 | 2 | 6 |
| Agent_2 | 4 | 0 | 5 | 3 |
| Agent_3 | 2 | 5 | 0 | 7 |
| Agent_4 | 6 | 3 | 7 | 0 |

Table 16. Distance of Agents to Object

| Agent to Object Distance | Object distance(m) |
|--------------------------|--------------------|
| Agent_1 | 2 |
| Agent_2 | 2 |
| Agent_3 | 3 |
| Agent_4 | 3 |

Table 17. Agents Trust Score Evaluation- For Experiment 2

Kindly refer to table sheet

$$V = \frac{1}{k \cdot \beta} = \frac{1}{0.2 \times 0.5} = 10 \text{ km (10,000 m)}$$

| Agent visibility due to weather conditions | | | | |
|--|---------------------|---------------------|------------------------|------------------------|
| | Agent_1 | Agent_2 | Agent_3 | Agent_4 |
| k | 0.2 | 0.2 | 0.3 | 0.3 |
| β | 0.5 | 0.5 | 0.5 | 0.5 |
| V | 10,000 m = 1 | 10,000 m = 1 | 6666.7 m = 0.70 | 6666.7 m = 0.70 |

- Solving for the predicted accuracy (A) of each agent.
 $w(A) = \alpha \cdot w(D) + \beta \cdot w(V) + \gamma \cdot w(T)$

| Agent Predicted accuracy (A) | | | | | |
|------------------------------|-------------|-------------|-------------|-------------|------------|
| Weight | Agent_1 | Agent_2 | Agent_3 | Agent_4 | Parameter |
| $\alpha = 0.3$ | 1 | 1 | 0.7 | 0.7 | Distance |
| $\beta = 0.3$ | 1 | 1 | 0.7 | 0.7 | Visibility |
| $\gamma = 0.4$ | 0.74 | 0.94 | 0.89 | 0.77 | Trust |
| A | 0.89 | 0.97 | 0.78 | 0.73 | |

$$P = a \cdot w(D) + b \cdot w(V) + c \cdot w(T) + d \cdot w(A)$$

Table 18. Experiment 2 Credibility Pooling Percentage For every Agent

| Agents Pooling accuracy (P) | | | | | |
|-----------------------------|-------------|--------------|-------------|--------------|---------------------|
| Weight | Agent_1 | Agent_2 | Agent_3 | Agent_4 | Parameter |
| $a = 0.2$ | 1 | 1 | 0.7 | 0.7 | Distance |
| $b = 0.2$ | 1 | 1 | 0.7 | 0.7 | Visibility |
| $c = 0.3$ | 0.7390 | 0.9425 | 0.895 | 0.7665 | Trust |
| $d = 0.3$ | 0.8956 | 0.97 | 0.778 | 0.7266 | Predictive accuracy |
| P | 0.89 | 0.974 | 0.78 | 0.728 | |

III. VALIDATION

To evaluate the performance of each drone, data collected under different environmental conditions were scrutinized. Image quality, resolution, and clarity were assessed to ascertain the drones' ability to capture accurate and detailed data. Flight stability and maneuverability were also evaluated to gauge the drones' performance in adverse weather conditions. Additionally, the drones' adaptability to changing environmental variables and their ability to maintain consistent data capture rates were considered. The results of the evaluation provided valuable insights into the drones' performance characteristics and their suitability for specific data extraction tasks under different conditions. As depicted in Fig. 10, alterations in drones' positions and environmental factors may enhance or diminish a drone's performance or accuracy in data pooling. In this scenario, there was 29% improvement for Agent_1 and Agent_4, while Agent_2 and Agent_3 experienced negligible changes.

In contrast to the work by C. Burnett et al. [61], which primarily focuses on abstract trust dynamics among agents, the experimental approach presented in this research places a stronger emphasis on assessing tangible performance metrics of agents, specifically drones. Rather than delving into theoretical frameworks, this approach prioritizes practical evaluations of drones' performance under varying environmental conditions [62], [63]. While Chogyal et al. primarily concentrates on developing a trust assessment model [63], the experimental methodology indirectly addresses trust by empirically evaluating the reliability and accuracy of drones' data capture capabilities. This shift

towards empirical assessment allows for a more concrete understanding of the drones' capabilities in real-world scenarios. While Huang et al. explores trust and reputation dynamics among agents [64], this research focuses on the practical implications of drone performance and reliability. By providing empirical insights, it contrasts with Huang et al.'s theoretical approach and offers valuable information for decision-making. Similarly, while Barber et al. address challenges in trust research [65], this study bridges theory and application by empirically assessing drone performance. Unlike Drawel et al.'s focus on formal verification [66], this research evaluates drones under empirical conditions, providing actionable insights. Finally, in contrast to G. Lu et al.'s review of computational trust models [67], this study emphasizes practical assessments of drone reliability for informed decision-making.

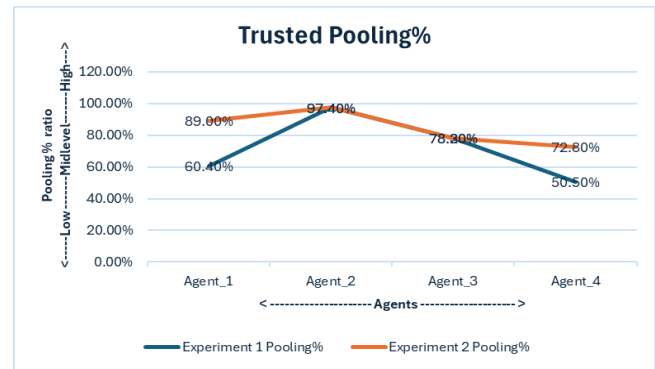


Fig. 10. Agents pooling percentage from both experiments

IV. CONCLUSION

In conclusion, this research provides valuable insights into performance coordination through communication, trust evaluation, and data pooling under various environmental conditions, with a focus on practical assessments such as delay in communication, image quality, flight stability, and data pooling. The contribution of this research lies in implementing a pre-assessment of cyber threats by analyzing potential attacks during communication delay checks. It also evaluates the agents' positions concerning a safe operating environment and ensures trusted communication by verifying both the source of received information and the validity of the transmitted data. This research does not address equipment malfunctions that could lead to environmental issues in the event of failure. Instead, it primarily focuses on control through trusted communication, ensuring reliability in data exchange. It also aims to minimize communication delays by dynamically switching between network topologies, providing an optimal approach to reducing latency in multi-agent systems. By indirectly addressing trust through the reliability and accuracy of agents' position, data capture, the study bridges the gap between theory and practical applications, offering actionable insights for informed decision-making in MAS deployment and advancing knowledge in the field of MAS technology. While this study evaluates the credibility of agents in MAS and communication delay, future research will expand on amending communication language or pattern for which agents can applied in more elaborating ways to enhance their coordination. This can be done through the integration or implementation of machine learning or artificial intelligence

model. By implementing an adequate language of communication, agents can retrieve information or translate information into needed mathematical coordinates. An advanced algorithms for communication and data evaluation will help extract deeper insights from drone-generated datasets, improving decision-making. Additionally, developing robust data transfer protocols will ensure faster, more reliable communication, addressing challenges like latency and bandwidth limitations, and enhancing real-time drone operations in applications such as surveillance, mapping, and environmental monitoring.

ACKNOWLEDGMENT

This research is based upon work supported by the ONR (Award No. N00014-22-1-2724) and Boeing. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the U.S. Government. Stefalo Acha is a student at North Carolina Agricultural and Technical State University, Greensboro, NC 27411 USA. (e-mail: sacha@aggies.ncat.edu). Sun Yi is an associate professor of Mechanical Engineering at North Carolina Agricultural and Technical State University, Greensboro, NC27411(email: ysi@ncat.edu).

REFERENCES

- [1] J. Hu, P. Bhowmick, I. Jang, F. Arvin and A. Lanzon, "A Decentralized Cluster Formation Containment Framework for Multirobot Systems," *IEEE Transactions on Robotics*, vol. 37, no. 6, pp. 1936-1955, 2021, <https://doi.org/10.1109/TRO.2021.3071615>.
- [2] J. Sumpter, C. Thomas, S. Yi and A. Kruger, "Analysis of the Effects of Communication Trust and Delay on Consensus of Multi-Agent Systems," *2019 IEEE 2nd Connected and Automated Vehicles Symposium (CAVS)*, pp. 1-6, 2019, <https://doi.org/10.1109/CAVS.2019.8887827>.
- [3] N. C. A. de Freitas and M. P. dos Santos Silva, "Multiagent system for image mining," *InTech*, 2017, <https://doi.org/10.5772/intechopen.69353>.
- [4] N. Nguyen, D. Nguyen, J. Kim, G. Rizzo and H. Nguyen, "Multi-Agent Data Collection in Non-Stationary Environments," *2022 IEEE 23rd International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, pp. 120-129, 2022, <https://doi.org/10.1109/WoWMoM54355.2022.00023>.
- [5] Y. Liu, J. Li, F. Fang and J. H. Park, "Further Studies on Sampled-Data Consensus of Multi-Agent Systems With Communication Delays," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 8, pp. 920-931, 2022, <https://doi.org/10.1109/TSIPN.2022.3220153>.
- [6] G. Wen, Z. Duan, W. Yu, G. Chen, "Consensus in multi-agent systems with communication constraints," *International Journal of Robust and Nonlinear Control*, vol. 22, no. 2, pp. 170-182, 2012, <https://doi.org/10.1002/rnc.1687>.
- [7] A. Amirkhani, A. H. Barshooi, "Consensus in multi-agent systems: a review," *Artificial Intelligence Review*, vol. 55, no. 5, pp. 3897-3935, 2022, <https://doi.org/10.1007/s10462-021-10097-x>.
- [8] G. Papoudakis, F. Christianos, A. Rahman, S. V. Albrecht, "Dealing with non-stationarity in multi-agent deep reinforcement learning," *arXiv*, 2019, <https://doi.org/10.48550/arXiv.1906.04737>.
- [9] J. Liu and Y. Tang, "Conflict data fusion in a multi-agent system premised on the base basic probability assignment and evidence distance," *Entropy*, vol. 23, no. 7, pp. 820, 2021, <https://doi.org/10.3390/e23070820>.
- [10] A. Stephen, R. Karthikeyan, C. Sowmiya, R. Raja, R. P. Agarwal, "Sampled-data controller scheme for multi-agent systems and its Application to circuit network," *Neural Networks*, vol. 170, pp. 506-520, 2024, <https://doi.org/10.1016/j.neunet.2023.11.059>.
- [11] J. Zhan and X. Li, "Consensus of sampled-data multi-agent networking systems via model predictive control," *Automatica*, vol. 49, no. 8, pp. 2502-2507, 2013, <https://doi.org/10.1016/j.automatica.2013.04.037>.
- [12] S. Huang, J. Layer, D. Smith, G. P. Bingham, Q. Zhu, "The effect of movement frequency on perceptual-motor learning of a novel bimanual coordination pattern," *Human Movement Science*, vol. 83, pp. 102958, 2022, <https://doi.org/10.1016/j.humov.2022.102958>.
- [13] M. A. Goodale, G. Króliczak and D. A. Westwood, "Dual routes to action: contributions of the dorsal and ventral streams to adaptive behavior," *Progress in Brain Research*, vol. 149, pp. 269-283, 2005, [https://doi.org/10.1016/S0079-6123\(05\)49019-6](https://doi.org/10.1016/S0079-6123(05)49019-6).
- [14] P. R. DeLucia, "Time-to-contact judgments of an approaching object that is partially concealed by an occluder," *Journal of Experimental Psychology: Human Perception and Performance*, vol. 30, no. 2, pp. 287, 2004, <https://psycnet.apa.org/doi/10.1037/0096-1523.30.2.287>.
- [15] C. Best, "The Influence of Ship Deck-Edge Lighting on Perception of Position and Movement during Helicopter Recovery," *Australian Government*, 2006, <https://apps.dtic.mil/sti/citations/ADA462950>.
- [16] M. Jang, A. Ahmed and G. Agha, "Efficient agent communication in multi-agent systems," *Software Engineering for Multi-Agent Systems III*, pp. 236-253, 2005, https://doi.org/10.1007/978-3-540-31846-0_14.
- [17] N. Borselius, "Multi-Agent System Security for Mobile Communication," *Computer Science, Engineering*, 2003, <https://www.semanticscholar.org/paper/Multi-agent-system-security-for-mobile-Borselius/cff632354c26393e86fd3ac16e5e0474464bf4c1>.
- [18] K. You and L. Xie, "Network Topology and Communication Data Rate for Consensusability of Discrete-Time Multi-Agent Systems," *IEEE Transactions on Automatic Control*, vol. 56, no. 10, pp. 2262-2275, 2011, <https://doi.org/10.1109/TAC.2011.2164017>.
- [19] J. Yu and L. Wang, "Group consensus in multi-agent systems with switching topologies and communication delays," *Systems & Control Letters*, vol. 59, no. 6, pp. 340-348, 2010, <https://doi.org/10.1016/j.sysconle.2010.03.009>.
- [20] S. Yi and A. G. Ulsoy, "Time-delayed vision-based DC motor control via rightmost eigenvalue assignment," *2014 American Control Conference*, pp. 5564-5569, 2014, <https://doi.org/10.1109/ACC.2014.6858677>.
- [21] S. Yi, S. Yu and J. H. Kim, "Analysis of neural networks with time-delays using the Lambert W function," *Proceedings of the 2011 American Control Conference*, pp. 3221-3226, 2011, <https://doi.org/10.1109/ACC.2011.5991085>.
- [22] K. F. Firoz, Y. Seong, S. Yi and Y. Chang, "A Preliminary Study on Human Trust in Pseudo-Real-Time Scenario through Electroencephalography and Machine Learning based Data Classification," *1st International Conference on Smart Mobility and Vehicle Electrification*, 2023, <https://doi.org/10.46254/EV01.20230061>.
- [23] D. Chilukuri, S. Yi and Y. Seong, "Development of Mobile Application for VRUs Using Computer Vision," *SoutheastCon 2018*, pp. 1-6, 2018, <https://doi.org/10.1109/SECON.2018.8479138>.
- [24] P. Shi and B. Yan, "A Survey on Intelligent Control for Multiagent Systems," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 1, pp. 161-175, 2021, <https://doi.org/10.1109/TSMC.2020.3042823>.
- [25] P. Akangah and M. A. Mukhtadir, "A study of elliptical delamination of a Sub laminate in a Quasi-Isotropic Composite Laminate: A fresh look at the solution method," *International Association of Journals and Conferences*, 2022, https://2022.iajc.org/wp-content/uploads/simple-file-list/uploaded-papers/proceeding-papers/014-M-22___A-study-of-Elliptical-Delamination-of-a-Sub-Laminate-in-a-Quasi-Isotropic-Composite___Akanah-Mukhtadir___CONFERENCE-PROCEEDINGS2.docx.
- [26] A. A. Tereda and S. Yi, "Predictive control of the KINOVA Gen3 robotic manipulator using a nonlinear model," *ASME International Mechanical Engineering Congress and Exposition*, 2024, <https://doi.org/10.1115/IMECE2023-112777>.
- [27] A. A. Tereda, "Path Planning and Sensing for Autonomous Control of Robot Manipulators," *North Carolina Agricultural and Technical State University*, 2021, <https://www.proquest.com/openview/3feb689bab3761b0c3f02d3fced6ece9/1?cbl=18750&diss=y&pq-origsite=gscholar>.

- [28] S. Acha, S. Yi, F. Ferguson, "Control of Leading-Edge Shock Train Using Deep Neural Network to Prevent Unstart," *Control Systems and Optimization Letters*, vol. 2, no. 3, pp. 285-294, 2022, <http://dx.doi.org/10.59247/csol.v2i3.160>.
- [29] B. Wang, S. Li, G. Battistelli, L. Chisci, W. Yi, "Principled information fusion for multi-view multi-agent surveillance systems," *ArXiv*, 2021, <https://doi.org/10.48550/arXiv.2105.10935>.
- [30] C. D. Schultz, "A trust framework model for situational contexts," *Proceedings of the 2006 International Conference on Privacy, Security and Trust: Bridge the Gap between PST Technologies and Business Services*, 2006, <https://doi.org/10.1145/1501434.1501494>.
- [31] P. S. P. Wong and S. O. Cheung, "Structural equation model of trust and partnering success," *Journal of Management in Engineering*, vol. 21, no. 2, pp. 70-80, 2005, [https://doi.org/10.1061/\(ASCE\)0742-597X\(2005\)21:2\(70\)](https://doi.org/10.1061/(ASCE)0742-597X(2005)21:2(70)).
- [32] J. Dearmon and K. Grier, "Trust and development," *Journal of Economic Behavior & Organization*, vol. 71, no. 2, pp. 210-220, 2009, <https://doi.org/10.1016/j.jebo.2009.02.011>.
- [33] D. J. Essin, "Patterns of trust and policy," *Proceedings of the 1997 Workshop on New Security Paradigms*, 1998, <https://doi.org/10.1145/283699.283738>.
- [34] P. R. DeLucia, "Critical roles for distance, task, and motion in space perception: Initial conceptual framework and practical implications," *Human Factors*, vol. 50, no. 5, pp. 811-820, 2008, <https://doi.org/10.1518/001872008X312297>.
- [35] M. Taddeo, "Modelling trust in artificial agents, a first step toward the analysis of e-trust," *Minds and Machines*, vol. 20, pp. 243-257, 2010, <https://doi.org/10.1007/s11023-010-9201-3>.
- [36] D. Asher, M. Garber-Barron, S. Rodriguez, E. Zaroukian and N. Waytowich, "Multi-Agent Coordination Profiles through State Space Perturbations," *2019 International Conference on Computational Science and Computational Intelligence (CSCI)*, pp. 249-252, 2019, <https://doi.org/10.1109/CSCI49370.2019.00051>.
- [37] S. Armah and S. Yi, "Altitude regulation of quadrotor types of UAVs considering communication delays," *IFAC-PapersOnLine*, vol. 48, no. 12, pp. 263-268, 2015, <https://doi.org/10.1016/j.ifacol.2015.09.388>.
- [38] S. Yi, S. Duan, P. W. Nelson, A. G. Ulsoy, "The Lambert W function approach to time delay systems and the LambertW_DDE toolbox," *IFAC Proceedings Volumes*, vol. 45, no. 14, pp. 114-119, 2012, <https://doi.org/10.3182/20120622-3-US-4021.00008>.
- [39] M. T. Matthews and S. Yi, "Neural Network Based Adaptive Flight Control of UAVs," *SoutheastCon 2021*, pp. 1-7, 2021, <https://doi.org/10.1109/SoutheastCon45413.2021.9401817>.
- [40] Y. Ito, M. A. S. Kamal, T. Yoshimura, S. Azuma, "Pseudo-perturbation-based broadcast control of multi-agent systems," *Automatica*, vol. 113, pp. 108769, 2020, <https://doi.org/10.1016/j.automatica.2019.108769>.
- [41] H. Zhu and M. Kirley, "Deep Multi-agent Reinforcement Learning in a Common-Pool Resource System," *2019 IEEE Congress on Evolutionary Computation (CEC)*, pp. 142-149, 2019, <https://doi.org/10.1109/CEC.2019.8790001>.
- [42] K. Garg and S. K. Nayar, "Detection and removal of rain from videos," *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, pp. I-I, 2004, <https://doi.org/10.1109/CVPR.2004.1315077>.
- [43] K. Garg and S. K. Nayar, "Vision and rain," *International Journal of Computer Vision*, vol. 75, pp. 3-27, 2007, <https://doi.org/10.1007/s11263-006-0028-6>.
- [44] S. Loussaief and A. Abdelkrim, "Machine learning framework for image classification," *2016 7th International Conference on Sciences of Electronics, Technologies of Information and Telecommunications (SETIT)*, pp. 58-61, 2016, <https://doi.org/10.1109/SETIT.2016.7939841>.
- [45] A. Nahid and Y. Kong, "Involvement of machine learning for breast cancer image classification: a survey," *Computational and Mathematical Methods in Medicine*, vol. 2017, no. 1, 2017, <https://doi.org/10.1155/2017/3781951>.
- [46] A. J. Edney and M. J. Wood, "Applications of digital imaging and analysis in seabird monitoring and research," *Ibis*, vol. 163, no. 2, pp. 317-337, 2021, <https://doi.org/10.1111/ibi.12871>.
- [47] H. Friedrich, M. Kaiser, O. Rogalla, and R. Dillmann, "Learning and communication in multi-agent systems," *Workshop on Learning in Distributed Artificial Intelligence Systems*, pp. 259-275, 1996, https://doi.org/10.1007/3-540-62934-3_53.
- [48] S. Li, W. Song, L. Fang, Y. Chen, P. Ghamisi and J. A. Benediktsson, "Deep Learning for Hyperspectral Image Classification: An Overview," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 9, pp. 6690-6709, 2019, <https://doi.org/10.1109/TGRS.2019.2907932>.
- [49] P. Xuan, V. Lesser and S. Zilberstein, "Communication decisions in multi-agent cooperation: Model and experiments," *Proceedings of the Fifth International Conference on Autonomous Agents*, 2001, <https://doi.org/10.1145/375735.376469>.
- [50] R. M. V. Eijk, F. S. D. Boer, W. V. D. Hoek and J. C. Meyer, "Open multi-agent systems: Agent communication and integration," *Intelligent Agents VI. Agent Theories, Architectures, and Languages*, pp. 218-232, 2000, https://doi.org/10.1007/10719619_16.
- [51] A. Singh and P. Singh, "Image classification: a survey," *Journal of Informatics Electrical and Electronics Engineering*, vol. 1, no. 2, pp. 1-9, 2020, <http://dx.doi.org/10.54060/JIEEE/001.02.002>.
- [52] V. A. Demin, D. B. Efimenko, D. V. Blinov, D. A. Komkova and V. R. Rogov, "Multi-Agent Approach to Freight Transportation Using Pooling Technology," *2021 Systems of Signals Generating and Processing in the Field of on Board Communications*, pp. 1-5, 2021, <https://doi.org/10.1109/IEEECONF51389.2021.9416077>.
- [53] Z. Ren and C. J. Anumba, "Multi-agent systems in construction—state of the art and prospects," *Automation in Construction*, vol. 13, no. 3, pp. 421-434, 2004, <https://doi.org/10.1016/j.autcon.2003.12.002>.
- [54] Y. Peng *et al.*, "A multi-agent system for enterprise integration," *International Journal of Agile Manufacturing*, vol. 1, no. 2, pp. 213-229, 1998, <https://ebiquity.umbc.edu/paper/html/id/322>.
- [55] A. Dorri, S. S. Kanhere and R. Jurdak, "Multi-Agent Systems: A Survey," *IEEE Access*, vol. 6, pp. 28573-28593, 2018, <https://doi.org/10.1109/ACCESS.2018.2831228>.
- [56] P. Dollar, Z. Tu, H. Tao and S. Belongie, "Feature Mining for Image Classification," *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1-8, 2007, <https://doi.org/10.1109/CVPR.2007.383046>.
- [57] J. De La Calleja and O. Fuentes, "Machine learning and image analysis for morphological galaxy classification," *Monthly Notices of the Royal Astronomical Society*, vol. 349, no. 1, pp. 87-93, 2004, <https://doi.org/10.1111/j.1365-2966.2004.07442.x>.
- [58] C. Lee, J. Lawry and A. Winfield, "Combining opinion pooling and evidential updating for multi-agent consensus," *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, pp. 347-353, 2018, <https://doi.org/10.24963/ijcai.2018/48>.
- [59] J. Xie and C. Liu, "Multi-agent systems and their applications," *Journal of International Council on Electrical Engineering*, vol. 7, no. 1, pp. 188-197, 2017, <https://doi.org/10.1080/22348972.2017.1348890>.
- [60] F. Wu, S. Zilberstein and X. Chen, "Online planning for multi-agent systems with bounded communication," *Artificial Intelligence*, vol. 175, no. 2, pp. 487-511, 2011, <https://doi.org/10.1016/j.artint.2010.09.008>.
- [61] C. Burnett, T. J. Norman and K. Sycara, "Trust decision-making in multi-agent systems," *Twenty-Second International Joint Conference on Artificial Intelligence*, 2011, <https://www.ijcai.org/Proceedings/11/Papers/031.pdf>.
- [62] M. Cheng *et al.*, "A general trust framework for multi-agent systems," *Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems*, 2021, <https://www.ifaamas.org/Proceedings/aamas2021/pdfs/p332.pdf>.
- [63] K. Chhogyal, A. Nayak, A. Ghose, H. K. Dam, "A value-based trust assessment model for multi-agent systems," *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, pp. 194-200, 2019, <https://doi.org/10.24963/ijcai.2019/28>.
- [64] H. Huang, G. Zhu and S. Jin, "Revisiting Trust and Reputation in Multi-agent Systems," *2008 ISECS International Colloquium on Computing, Communication, Control, and Management*, pp. 424-429, 2008, <https://doi.org/10.1109/CCCM.2008.122>.
- [65] K. S. Barber, K. Fullam and J. Kim, "Challenges for trust, fraud and deception research in multi-agent systems," *Trust, Reputation, and Security: Theories and Practice*, pp. 8-14, 2003, https://doi.org/10.1007/3-540-36609-1_2.

- [66] N. Drawel, J. Bentahar, A. Laarej and G. Rjoub, "Formal verification of group and propagated trust in multi-agent systems," *Autonomous Agents and Multi-Agent Systems*, vol. 36, no. 1, p. 19, 2022, <https://doi.org/10.1007/s10458-021-09542-6>.
- [67] G. Lu *et al*, "A review on computational trust models for multi-agent systems," *The Open Information Science Journal*, vol. 2, pp. 18-25, 2009, [https://eprints.hud.ac.uk/id/eprint/4102/1/18TOISCIJ\[1\]lu2.pdf](https://eprints.hud.ac.uk/id/eprint/4102/1/18TOISCIJ[1]lu2.pdf).