UDC 004.8 https://doi.org/10.20535/2786-8729.6.2025.331602

VISION-BASED NEIGHBOR SELECTION METHOD FOR OCCLUSION-RESILIENT UNCREWED AERIAL VEHICLE SWARM COORDINATION IN THREE-DIMENSIONAL ENVIRONMENTS

Oleksii Smovzhenko *
National Technical University of Ukraine
“Igor Sikorsky Kyiv Polytechnic Institute”, Kyiv, Ukraine
https://orcid.org/0009-0006-0500-5005

Andrii Pysarenko
National Technical University of Ukraine
“Igor Sikorsky Kyiv Polytechnic Institute”, Kyiv, Ukraine
https://orcid.org/0000-0001-7947-218X

*Corresponding author: alekseismovzhenko@gmail.com

Uncrewed aerial vehicle (UAV) swarms provide superior scalability, reliability, and efficiency compared to individual UAVs, enabling transformative applications in search and rescue, precision agriculture, environmental monitoring, and urban surveillance. However, their dependence on Global Navigation Satellite Systems (GNSS) and wireless communication introduces vulnerabilities like signal loss, jamming, and scalability constraints, particularly in GNSS-denied environments. This study advances swarm robotics by developing a novel neighbor selection method for occlusion-resilient, vision-based coordination of UAV swarms in three-dimensional (3D) environments, addressing the problem of visual occlusions that disrupt decentralized flocking. Unlike prior research focusing on planar settings or communication-dependent systems, we model swarm coordination as an artificial potential field problem. Additionally, we evaluate performance through metrics like minimum nearest neighbor distance (collision avoidance), alignment (velocity synchronization), and union (cohesion). Using simulations in point mass and realistic quadcopter dynamics (Gazebo with PX4) environments, we assess swarm behavior across dense, default, and sparse configurations. Our findings reveal that occlusions degrade alignment (below 0.9) and distances (below 0.5 m) in dense swarms exceeding 70 agents, increasing collision risks. Our novel method, incorporating metric, topographic, and Delaunay strategies, mitigates these effects. Topographic selection achieves high alignment (above 0.9) in small swarms (up to 50 agents), while Delaunay ensures perfect cohesion (union = 1) and robust alignment across all swarm sizes. Validation in simulations confirms these results. Furthermore, our method enables communication-free coordination that matches or surpasses communication-enabled performance, with topographic selection outperforming (alignment above 0.9 vs. 0.85) in small swarms and Delaunay excelling in larger ones. This result eliminates the need for inter-agent communication, enhancing resilience and bandwidth efficiency. These findings establish a scalable, infrastructure-independent framework for UAV swarms, with practical value for autonomous operations in complex, occlusion-prone environments.

Keywords: UAV swarm, vision-based localization, formation control, decentralized coordination, artificial potential field.

1. Introduction

Swarm robotics, a dynamic field within autonomous systems, focuses on coordinating multiple robots to achieve collective goals, drawing inspiration from natural systems like bird flocks and fish schools. Uncrewed aerial vehicles (UAVs) have evolved significantly since their conceptual origins in the early 20th century. Initial developments focused on military applications, such as reconnaissance and target practice, with mass production emerging mid-century. Advancements in electronics enabled the integration of compact, lightweight components, facilitating sophisticated sensors and communication systems for long-distance remote control and real-time data transmission. The advent of powerful computing units marked a crucial change, allowing UAVs to follow preprogrammed paths autonomously. Breakthroughs in materials science introduced lightweight, durable composites, enhancing flight endurance and enabling diverse applications. Examples of these applications include agriculture, infrastructure inspection, environmental monitoring, disaster management, humanitarian aid, logistics, and entertainment, reflecting the growing relevance of aerial robotics in addressing modern problems.

UAV swarms, defined as groups of UAVs operating cooperatively to achieve shared objectives, represent a significant leap beyond individual UAV capabilities. Swarms use collective behavior to enhance efficiency, offering advantages such as scalability, expanded area coverage, improved reliability, and the ability to execute complex tasks at reduced costs. In swarm robotics, this decentralized approach, where each UAV observes its local environment to perform localized tasks, contributes to global goals, making swarms ideal for dynamic, real-world applications requiring robust coordination. This distributed approach enables swarms to adapt dynamically to changing conditions, underscoring their potential in scenarios like disaster response and precision agriculture, where adaptability and scalability are important.

The scientific problem addressed in this study lies in achieving robust coordination in UAV swarms using vision-based localization, particularly under visual occlusions in three-dimensional (3D) environments. The versatility of UAV swarms has driven their adoption across numerous domains. In agriculture, swarms provide high-resolution aerial imagery for crop monitoring and precision farming, optimizing resource use over vast areas. In construction and mining, they enable progress tracking and 3D mapping, improving planning and safety. Infrastructure inspection benefits from swarms' ability to navigate and assess complex structures autonomously. For healthcare and humanitarian aid, swarms deliver medical supplies and support emergency responses in remote or disaster-affected regions. Public safety and security operations utilize swarms for search and rescue missions and enhanced surveillance in challenging terrains. In logistics, swarms facilitate last-mile delivery solutions, particularly in regions with limited infrastructure. Telecommunications use swarms as aerial networks for temporary coverage, while disaster mitigation employs them for environmental monitoring and hazard detection, highlighting the topic's relevance to modern societal needs [1, 2].

Despite these advancements, decentralized UAV swarms face significant problems in navigation and coordination, particularly in achieving occlusion-resilient vision-based localization. Most current implementations rely heavily on the Global Navigation Satellite System (GNSS), such as the Global Positioning System (GPS), for localization and navigation. These systems are susceptible to positional inaccuracies, signal jamming, and spoofing, which can disrupt swarm operations. Additionally, wireless inter-agent or base station communication introduces scalability constraints and vulnerabilities, limiting swarms' effectiveness in GNSS-denied environments like indoor spaces or cluttered urban settings. The problem of visual occlusions – where neighboring agents or environmental obstacles block line-of-sight – further complicates vision-based localization, as it restricts the detection of nearby UAVs, important for maintaining cohesive swarm behavior.

This study addresses the pressing need for occlusion-resilient, infrastructure-independent UAV swarm coordination, a topic of growing importance in swarm robotics due to increasing demands for autonomous systems in complex, dynamic environments. Vision-based localization, using onboard cameras and computer vision algorithms, enables UAVs to estimate neighbor positions and orientations without GNSS or extensive communication, offering a solution for decentralized flocking. However, occlusions in dense formations reduce the effectiveness of visual sensing, necessitating innovative methods like neighbor selection to ensure robust performance. The relevance of this scientific topic stems from its potential to enable scalable, autonomous swarm deployments. Such swarms could be utilized in various domains, ranging from disaster response to urban surveillance, where traditional navigation systems falter. In conclusion, studying occlusion-resilient vision-based swarm coordination addresses modern problems in automation, resilience, and infrastructure independence across diverse fields.

2. Literature review and problem statement

UAV swarms have become a center of interest in robotics research due to their ability to perform complex tasks cooperatively, such as environmental monitoring, disaster response, and infrastructure inspection. Traditional swarm navigation relies heavily on the GNSS, such as GPS, to provide positional data for localization and coordination. However, GNSS signals are prone to inaccuracies, especially in dense formations where small positional errors can lead to misalignment or collisions. These signals are also susceptible to external disruptions, such as jamming or spoofing, which can compromise the entire swarm's operation [3]. Furthermore, decentralized swarms depend on wireless communication for inter-agent coordination or base station interaction, introducing significant time lags. These lags stem from a combination of GNSS's low update rates, control loop latency, and communication delays, which can destabilize swarm motion, causing oscillations and increasing the likelihood of collisions in dense configurations [3]. As the number of drones in a swarm increases, the volume of data transmitted for coordination grows quadratically, potentially oversaturating communication channels and leading to data loss or delays [4]. While advanced hardware, such as high-bandwidth transceivers or more precise GNSS receivers, can partially mitigate these issues, such solutions increase drone weight or cost, reducing flight duration and operational efficiency.

To address these limitations, researchers have explored alternative sensory modalities for localization, with vision-based approaches emerging as a promising solution. Vision-based localization enables UAVs to estimate the position and orientation of neighboring agents using onboard cameras and computer vision algorithms, eliminating the need for external infrastructure like GNSS. This method supports fully decentralized coordination, allowing swarms to operate autonomously in GNSS-denied environments, such as indoor spaces, urban canyons, or cluttered outdoor settings. By relying on onboard sensing, vision-based systems reduce dependence on wireless communication, enhancing resilience against signal interference or loss [5]. Vision-based localization can leverage various spectral bands, including infrared (IR), ultraviolet (UV), and visible light. For example, researchers have developed systems using active IR-coded emitters mounted on drones to facilitate localization [5]. These emitters allow drones to identify neighbors through unique IR patterns, but their effectiveness is limited outdoors during daylight due to solar radiation interference across the IR spectrum.

An alternative approach, the Ultraviolet-based Visual Localization System (UVDAR), utilizes the UV spectrum to enable communication-free swarm coordination without reliance on GNSS or radio communication [6, 7]. UVDAR's use of the UV band, which is largely free from natural interference, makes it suitable for outdoor operations at any time of day, unlike IR-based methods. Experimental results have demonstrated UVDAR's ability to maintain robust localization in various environments, supporting applications like search and rescue or agricultural monitoring [6]. However, UVDAR requires specialized hardware, including UV markers and sensors, which add weight to drones, reducing flight endurance and limiting scalability for large swarms. To overcome this, a fully vision-based system using omnidirectional cameras and convolutional neural networks (CNNs) has been proposed, eliminating the need for additional markers [8]. This system processes visual data to estimate neighbor positions, demonstrating resilience under diverse lighting conditions and complex backgrounds in outdoor experiments. Such advancements highlight the potential of vision-based localization for autonomous, infrastructure-independent swarm operations.

Despite these advances, vision-based localization introduces the issue of visual occlusions, where other drones or environmental obstacles block the line of sight between agents. This problem is particularly pronounced in dense swarms, where frequent occlusions can disrupt localization accuracy and swarm cohesion. Researchers have addressed this by limiting the number of neighbors considered for coordination, using strategies like Voronoi diagrams to select neighbors based on spatial proximity [9]. These strategies improve performance under occlusions by reducing computational complexity and focusing on relevant agents. However, these studies were conducted in planar (2D) environments, which do not fully represent the spatial (3D) dynamics of most real-world UAV swarm applications, where movement occurs within height-constrained 3D spaces. The optimal number of neighbors for effective flocking has been investigated, with findings suggesting that 7–10 neighbors provide a balance between network complexity and convergence quality [10]. Topological neighbor selection, where each agent interacts with a fixed number of neighbors based on proximity, has been shown to outperform metric-based selection, which relies on a distance threshold, in achieving stable flocking [11]. These studies also indicate that both topological and metric interactions require a threshold (e.g., neighbor count or distance) to achieve an ordered state. However, they do not explore whether such strategies lead to swarm fragmentation, where subgroups of drones lose connectivity, reducing overall cohesion.

The body of research on UAV swarm flocking has made significant progress in addressing localization and coordination issues, but several limitations persist, particularly for vision-based systems operating in spatial environments. Most prior studies, including those employing Voronoi diagrams or topological neighbor selection, focus on planar (2D) environments, which simplify the dynamics of swarm motion. As a result, they fail to capture the complexities of real-world deployments where drones navigate in three-dimensional spaces, often constrained by minimum and maximum altitudes [9–11]. This planar focus overlooks the impact of visual occlusions in dense 3D formations, where drones frequently obstruct each other’s line of sight, potentially degrading alignment, increasing collision risks, and disrupting cohesive motion. The lack of spatial analysis limits the applicability of these findings to practical scenarios, such as infrastructure inspection or disaster response, where 3D navigation is essential.

Additionally, existing vision-based localization systems often rely on specialized hardware, such as IR-coded emitters or UV markers, which increase drone weight and reduce flight endurance, posing scalability constraints for large swarms [5–7]. While CNN-based systems eliminate the need for markers, their computational complexity can hinder real-time processing, especially for resource-constrained drones in large-scale deployments [8]. These hardware and computational demands highlight the need for lightweight, efficient vision-based solutions that can operate without additional equipment or excessive processing power.

Another underexplored area is the impact of neighbor selection strategies on swarm cohesion in 3D environments. Prior work suggests that limiting neighbors to 7–10 agents balance performance and complexity. However, these studies do not assess whether such strategies cause swarm fragmentation, where subgroups of drones become disconnected, undermining collective behavior [10, 11]. This is particularly relevant in dense swarms, where frequent occlusions may exacerbate fragmentation risks. Furthermore, the role of inter-agent communication in vision-based swarms remains insufficiently studied. While many systems assume some level of communication for coordination, eliminating this dependency could free bandwidth for other tasks, such as data transmission for environmental monitoring, enhancing overall swarm efficiency.

Existing occlusion models, such as those using ray casting, are computationally intensive, making them impractical for large swarms simulations where real-time processing is essential [9]. A lightweight occlusion model tailored for 3D environments could enable more efficient simulations and facilitate real-world deployments. Moreover, prior studies often assume idealized conditions, such as uniform drone shapes or simplified dynamics, which may not reflect the complexities of real UAVs with varied geometries or realistic flight constraints.

Despite advancements in swarm robotics, current research often overlooks the impact of visual occlusions on the performance of vision-based UAV swarms in 3D environments, particularly in terms of alignment, collision avoidance, and cohesion. A critical analysis reveals that 3D spatial settings, which are essential for real-world applications, introduce unique problems due to occlusions that disrupt vision-based localization. Consequently, the effectiveness of different neighbor selection strategies (metric, topological, and Delaunay) in mitigating these occlusions remains unexplored, as does the potential for achieving communication-free performance comparable to that of communication-enabled swarms. The unresolved problem is the lack of methods to ensure occlusion-resilient, decentralized coordination in 3D UAV swarms, limiting their scalability and reliability in complex environments. This gap justifies the purpose of the study to investigate vision-based neighbor selection strategies that enhance swarm performance in 3D settings without relying on inter-agent communication, addressing needs for autonomous, infrastructure-independent operations.

3. The aim and objectives of the study

This study aims to develop a method for occlusion-resilient, vision-based coordination of UAV swarms in 3D environments, addressing the unresolved problem of visual occlusions that disrupt decentralized flocking in GNSS-denied settings. Current swarm systems often rely on GNSS and wireless communication, which are vulnerable to signal loss, jamming, spoofing, and scalability constraints, particularly in complex environments such as indoor spaces or cluttered urban areas. These limitations hinder reliable, autonomous operation, especially in dense formations where visual occlusions from neighboring UAVs impair line-of-sight localization, reducing alignment and increasing collision risks. Our goal is to devise a neighbor selection method that ensures cohesive, collision-free swarm flocking without external dependencies, enhancing applicability to real-world scenarios. By tackling occlusions in 3D settings, this method aims to enable scalable, infrastructure-independent UAV swarms, validated through experimental data and analytical conclusions.

To achieve this aim, we define the following objectives:

– evaluate performance improvements from using a neighbor selection method, including metric, topographic, and Delaunay strategies, and their effectiveness in mitigating visual occlusion problem in vision-based UAV swarms at different density levels (dense, default, sparse).

– assess the performance of vision-based swarms using the optimal neighbor selection method against communication-enabled swarms in 3D environments, yielding analytical conclusions on whether communication-free coordination achieves comparable performance, enhancing scalability and resilience.

These objectives address the problem by providing a practical method and empirical evidence to support decentralized, occlusion-resilient flocking, thereby directly contributing to advancements in swarm robotics.

4. The study materials and methods of vision-based UAV swarm flocking and performance evaluation in 3D environment

4.1. Preliminary notations

We present a comprehensive review of a vision-based UAV swarm flocking method, along with an overview of performance metrics and simulation environments. These are used to simulate and evaluate vision-based UAV swarm flocking in a 3D environment, focusing on navigation toward a common goal while maintaining cohesive, collision-free motion. Our method encompasses preliminary notations to define agent relationships, a detailed flocking algorithm for coordinated motion, neighbor selection strategies to manage occlusions, and performance metrics to evaluate outcomes. Additionally, two simulation environments are described that were used in our experiments to strike a balance between computational efficiency and realism.

To model the swarm and its dynamics, we represent the swarm as a set of N homogeneous agents, each labeled by $i \in \mathrm{A}$. The set excluding agent i is denoted as $\mathrm{A}_i = \mathrm{A} \setminus \{i\}$, capturing all other agents in the swarm. Each agent's state is defined by its position and velocity, expressed as $p_i, v_i \in \mathbb{R}^3$, where p_i represents the agent's coordinates in 3D space, and v_i represents its velocity vector. The relative position of agent j with respect to agent i is calculated as:

$$r_{ij} = p_j - p_i . \qquad (1)$$

This vector quantifies the spatial relationship between agents, essential for vision-based localization. The distance between agents i and j is computed using the Euclidean norm:

$$d_{ij} = \left\| r_{ij} \right\| . \qquad (2)$$

This distance metric underpins neighbor selection and collision avoidance. We model the swarm as a directed graph, with vertices representing agents and edges indicating adjacency (denoted $i \sim j$). This graph is represented by an adjacency matrix A_{ij} of size $N \times N$, where entries are 1 if $i \sim j$ and 0 otherwise, capturing the connectivity structure of the swarm. Agent speeds are calculated at each time step k, though we omit this notation in subsequent sections for simplicity. The set of neighbors for agent i is denoted $|N_i| \in \mathrm{A}_i$, representing agents within its perception range. These notations provide a mathematical foundation for modeling swarm interactions and evaluating performance in a 3D environment.

4.2. Flocking algorithm

We design a robust flocking algorithm to enable UAV swarms to navigate toward a specified goal in a 3D environment while maintaining cohesive, collision-free motion. This algorithm should eliminate reliance on GNSS or inter-agent communication. This algorithm uses vision-based localization, using onboard cameras and computer vision to estimate neighbor positions and orientations, addressing the problems of visual occlusions in dense formations. By adapting Reynolds' flocking rules – separation, alignment, and cohesion – within an artificial potential field (APF) framework, we ensure decentralized coordination suitable for scalable swarms.

Our objective is to facilitate coordinated motion where each UAV maintains safe distances from neighbors, aligns its velocity with nearby agents, and moves collectively toward a common target without fragmenting into subgroups. To achieve this, we design a flocking algorithm using APF approach, incorporating Reynolds' flocking rules of separation, alignment, and cohesion. The velocity command for agent i is computed as the sum of social and migration components:

$$v_i = v_i^{\mathrm{soc}} + v_i^{\mathrm{mig}}, \tag{3}$$

where social velocity v_i^{soc} governs interactions with neighboring agents, ensuring cohesion and collision avoidance. It combines attractive forces that draw agents together to maintain swarm unity with repulsive forces that prevent collisions by ensuring adequate separation. These forces are derived from the APF, where potential fields model the influence of nearby agents based on their relative positions, as defined by the notations in Section 4.1. The migration velocity v_i^{mig} drives the swarm toward the goal, ensuring purposeful collective motion. By summing these components, the algorithm enables each UAV to make decentralized decisions based on local visual observations, reducing dependency on external infrastructure and supporting operations in GNSS-denied environments like indoor spaces or cluttered urban areas.

To ensure the velocity commands are executable in a physical environment, we normalize the resultant velocity to respect the drone's maximum speed limit:

$$\tilde{v}_i = \frac{v_i}{\|v_i\|} \min\left(\|v_i\|, v^{\max}\right), \tag{4}$$

where $v^{\max}$ represents the maximum allowable speed of the UAV, typically set to 1 m/s to reflect common quadcopter capabilities. The normalization process scales the velocity vector to maintain its direction while capping its magnitude at $v^{\max}$, ensuring that commands are feasible for physical drones with finite acceleration and thrust. This step enhances the algorithm's practicality, allowing seamless integration with real-world UAV control systems, such as those simulated in the Gazebo environment with PX4 integration (Section 4.5).

Cohesion and collision avoidance within the swarm can be achieved with a combination of attractive and repulsive potential. The attractive potential drives cohesion, encouraging agents to converge toward the average position of their neighbors, thereby preserving swarm unity. Conversely, the repulsive potential ensures separation, generating forces that prevent collisions by pushing agents away from nearby neighbors. The social velocity component integrates cohesion and separation forces and is expressed as:

$$v_i^{\mathrm{soc}} = k^{\mathrm{coh}} \frac{1}{|N_i|} \sum_{j \in N_i} r_{ij} - k^{\mathrm{sep}} \sum_{j \in N_i} \frac{r_{ij}}{\left\| r_{ij} \right\|^2}, \tag{5}$$

where k^{coh} is the cohesion gain regulating the attractive force, and k^{sep} is the separation gain controlling the repulsive force. We intentionally avoid scaling the separation speed component to maintain consistency across experiments, eliminating the need for fine-tuning gains for different swarm sizes or neighbor selection strategies (e.g., metric, topological, Delaunay). This approach enhances the algorithm's robustness and simplifies its application across diverse scenarios, ensuring reliable performance in both simulation and potential real-world deployments.

To enable goal-oriented navigation, we incorporate a migration velocity component that directs the swarm toward a specified target, ensuring purposeful collective motion. The migration velocity is defined as:

$$v_i^{\mathrm{mig}} = k^{\mathrm{mig}} \frac{r^{\mathrm{mig}}}{\left\| r^{\mathrm{mig}} \right\|}, \tag{6}$$

where r^{mig} is the relative position vector from agent to the goal position, and k^{mig} is the migration gain that regulates the strength of the goal-directed pull. This component ensures each UAV adjusts its trajectory to approach the target, aligning with the swarm's global objective while maintaining local coordination through the social velocity. The migration gain is carefully tuned to balance goal-directed movement with formation stability, preventing the swarm from dispersing or losing cohesion during navigation.

4.3. Neighbor selection

To manage visual occlusions and optimize computational efficiency, we implement neighbor selection strategies to limit the set of agents considered as neighbors, moving away from all-to-all connections. Prior research indicates that effective flocking does not require all agents to be neighbors, allowing us to reduce processing demands while maintaining performance [9–11]. We explore three strategies: metric, topological, and Delaunay triangulation-based, each designed to balance localization accuracy and swarm cohesion in the presence of occlusions.

The metric neighbor selection strategy selects agents within a fixed distance threshold, representing the drone's perception or communication range:

$$N_i^{\mathrm{metric}} = \left\{ j \in \mathrm{A}_i \mid d_{ij} \le r^{\max} \right\}, \tag{7}$$

where $r^{\max}$ defines the perception radius, ensuring only nearby agents are considered. This approach simplifies localization by focusing on spatially close neighbors, reducing the impact of distant occlusions.

The topological neighbor selection strategy selects the n nearest agents, regardless of distance, using:

$$N_i^{\mathrm{topo}} = \left\{ n - \arg\min d_{ij},\, j \in \mathrm{A}_i \right\}. \tag{8}$$

The $n - \arg\min$ operator identifies the n closest neighbors, providing a computationally efficient method that maintains stable swarm performance when n is appropriately chosen, as supported by prior studies [10, 11]. This strategy ensures consistent neighbor counts, mitigating occlusion effects in dense formations.

Researchers proposed a neighbor selection strategy using Voronoi diagrams, designating agents as neighbors if their regions share a common boundary [9]. However, constructing 3D Voronoi diagrams in spatial environments demands substantial computational resources. In our experiments, we observed that for small swarms $(n < 20)$, alignment sharply declines due to inadequate neighbor connections. Instead, we adopted Delaunay triangulation, computed efficiently as part of Voronoi diagram algorithms. This approach matches Voronoi performance for large swarms while significantly enhancing alignment in smaller swarms.

The Delaunay triangulation-based strategy selects neighbors based on shared ridges in a 3D triangulation of agent positions:

$$N_i^{\mathrm{del}} = \left\{ j \in \mathrm{P} \mid \mathrm{edge}(i, j) \text{ exists in } \mathrm{T}\,(\mathrm{P}\,) \right\}, \tag{9}$$

where P is the set of agent positions, and T is the Delaunay triangulation. This method typically yields up to 12 neighbors in 3D space, offering a spatially balanced selection that enhances cohesion.

The vision-based neighbor selection strategy identifies agents within a defined perception radius, akin to the metric selection approach, but excludes those partially obstructed by others. Detecting visual occlusions in 2D environments is straightforward, yet in 3D spaces, it presents a tough problem. Conventional methods like ray casting, common in computer graphics, demand extensive computational resources, potentially restricting swarm size in our simulations. To address this, we devised a streamlined, lightweight model for 3D visual occlusion, enabling experiments with larger swarms.

We consider agent j invisible to agent i if agent k occludes agent j, even partially, from agent i 's viewpoint. Each agent is modeled as a sphere with a fixed radius. By representing agents as spheres, we simplify the 3D problem into a 2D one by projecting the centers of three agents i, j, k onto a plane containing these points. This projection transforms agent spheres into circles of equal radius on the plane, allowing us to assess occlusion. We determine if agent k blocks agent j from agent i's perspective by verifying if the sum of their angular half-sizes exceeds their angular separation, but only when agent k is closer to i then j. This evaluation, repeated for all trios of agents, is less resource-intensive than ray casting, facilitating rapid simulations for robust statistical analysis. The model is expressed as:

$$N_i^{\mathrm{visual}} = \left\{ j \neq k \in N_i^{\mathrm{metric}} \mid \neg\left(\theta_{ij} + \theta_{ik} > \alpha_{ijk} \wedge d_{ik} < d_{ij} \right) \right\}, \tag{10}$$

where θ_{ij} and θ_{ik} denote the angular half-sizes of agents j and k from agent i's perspective, and α_{ijk} represents their angular separation. This approach captures visual constraints, supporting large-scale simulations with minimal computational overhead compared to ray casting. Collectively, these strategies enable us to evaluate how neighbor selection alleviates occlusion effects in vision-based swarms.

4.4. Swarm performance metrics

To evaluate the effectiveness of our flocking algorithm and neighbor selection strategies, we define three performance metrics: minimum nearest neighbor distance, alignment, and union. These metrics assess whether the swarm achieves collision-free, aligned, and cohesive navigation toward the goal, calculated at each discrete time step k.

The minimum nearest neighbor distance measures the smallest distance between any two agents, indicating collision avoidance:

$$d^{\min} = \min_{i \neq j} d_{ij}\,. \tag{11}$$

We consider a collision to have occurred if $d^{\min} < 2r$, where r is agent radius, ensuring safe separation during migration. This metric is important for evaluating the swarm's ability to maintain safe distances in dense formations.

The alignment metric quantifies how closely the swarm's agents move in the same direction:

$$\phi^{\text{align}} = \frac{1}{N(N-1)} \sum_{i \neq j} \frac{v_i \cdot v_j}{\|v_i\| \|v_j\|}. \tag{12}$$

A value of 1 indicates perfect alignment (all agents moving in the same direction), while 0 indicates complete disorder. A value of 0.9 is considered sufficient for effective flocking, reflecting synchronized motion. This metric helps assess the swarm's ability to maintain coordinated movement under occlusion constraints.

The union metric evaluates swarm cohesion, indicating whether the swarm moves as a single unit:

$$\phi^{\text{union}} = 1 - \frac{n^{\text{comp}} - 1}{N - 1}, \tag{13}$$

where n^{comp} is the number of connected components in the adjacency matrix. A value of 1 signifies a fully cohesive swarm, while 0 indicates fragmentation into isolated agents. This metric ensures the swarm remains connected, a major factor for collective tasks. These metrics collectively provide a comprehensive evaluation of swarm performance across varying conditions.

4.5. Simulation environments and drone models

We employ two simulation environments to balance computational efficiency and realistic dynamics. The first is the point mass environment implemented in pure Python, where agents are modeled as material points governed by simplified kinematics. Each agent's model can be described as a set of coordinates and a velocity vector in three-dimensional space. This environment does not account for aerodynamics, inertia, or actuator constraints. Control commands are directly translated into velocity updates at each discrete timestep. This type of abstraction significantly reduces computational requirements by simplifying flight dynamics. In this environment, the motion of agent i at step $k+1$ is described as:

$$p_i^{k+1} = p_i^k + v_i^k \cdot \Delta t. \tag{14}$$

This environment supports large-scale simulations with up to 150 agents, thanks to its lower computational demands, and is particularly suited for performance analysis of flocking algorithms and various neighbor selection strategies. Simulations can be performed without a graphical user interface.

Alternatively, the second environment, powered by Gazebo (Gazebo Classic), offers a physics-based simulation. Considering three-dimensional maneuverability, target applications mentioned in Section 1, and market availability, a quadrotor was selected as the drone type for our research. Although Gazebo supports various quadrotor models, the iris model (provided with PX4 Autopilot for Gazebo) was chosen because it closely resembles common quadrocopter models available on the market in terms of technical specifications. This model represents a drone with an approximate mass of 1.5 kg and a rotor arm length of 0.25 m. The quadrotor simulated in the Gazebo environment is subjected to realistic aerodynamics, inertial effects, and actuator dynamics. As a result, an agent cannot achieve the desired velocity instantaneously, unlike in the point mass dynamics environment, which affects swarm performance and may introduce possible collisions and disorder in the swarm. Control commands generated by the flocking algorithm are sent to the autopilot software (PX4), which interprets them within the flight control stack. This combination of tools produces a more realistic representation of actual UAV behavior, including delays and thrust saturation. An empty map with only ground and other agents, and no obstacles, has been selected for the experiments. Although this software allows for more realistic experiments, it requires higher computational resources. As a result, we were able to perform simulations in this environment for smaller swarms (up to 70 agents); beyond this, simulations become unstable. Although Gazebo and PX4 run asynchronously, synchronization can be achieved using the lockstepping feature, ensuring fair comparisons across swarm sizes. Using MAVSDK-Python, we emulated a ground control station to send velocity commands and receive telemetry. This simulation was deployed on a PC with Windows 11, running Ubuntu 20.04 through WSL2. PX4 software and the ground control station were deployed separately for each agent. In practice, this setup works as follows: the simulation is executed step by step, with a pause between each step. During these pauses, velocity commands are calculated for each agent using the proposed algorithms. Next, these commands are sent to the autopilot software for execution. At this point, the simulation is resumed and executed for Δt and then paused again. This process repeats until simulation completion, mitigating computational load increases with swarm size.

The key difference between environments lies in the fidelity of the flight model, computational demands, and scalability. The point mass dynamics environment ignores aerodynamics and software control loop side effects, prioritizing scalability, simulation duration, and computational efficiency, allowing us to perform large sets of experiments with large swarms. On the other hand, Gazebo incorporates realistic flight physics and actuator dynamics, which results in more realistic simulations at the cost of increased computational demand and smaller swarm sizes. These environments collectively enable the evaluation of vision-based flocking under a range of conditions, from idealized to realistic scenarios.

4.6. Experimental design overview

Our method integrates a cohesive set of components to evaluate the performance of vision-based UAV swarms in dense, spatial (3D) environments, addressing the impact of visual occlusions and the feasibility of communication-free flocking. We combine precise mathematical notations, a robust flocking algorithm, strategic neighbor selection, comprehensive performance metrics, carefully chosen experimental parameters. Finally, we use dual simulation environments to create a systematic framework for studying decentralized swarm behavior. This overview synthesizes these elements, illustrating how they work together to achieve the study's objectives of assessing occlusion effects, evaluating neighbor selection strategies, and comparing vision-based and communication-enabled swarms.

The preliminary notations (Section 4.1) establish a mathematical foundation by defining agent positions, velocities, and relationships as a directed graph, enabling us to model spatial interactions and connectivity. These notations underpin the flocking algorithm (Section 4.2), which uses an APF approach based on Reynolds' rules of separation, alignment, and cohesion. The algorithm computes velocity commands that balance social interactions (cohesion and collision avoidance) with goal-directed migration, ensuring drones move as a cohesive unit toward a target while avoiding collisions. By normalizing velocities to respect physical constraints, we ensure the algorithm is practical for real-world UAVs.

Neighbor selection strategies (Section 4.3) are central to managing visual occlusions, a primary focus of this study. We implement metric, topological, and Delaunay triangulation-based strategies to limit the number of neighbors considered, reducing computational demands and mitigating occlusion effects in dense formations. The novel vision-based neighbor selection, supported by a lightweight 3D occlusion model, simulates realistic visual constraints by identifying occluded agents, enabling scalable simulations without the computational burden of ray casting. These strategies allow us to test how different neighbor counts and selection methods influence swarm performance under varying conditions.

Performance metrics (Section 4.4), such as nearest neighbor distance, alignment, and union, provide a comprehensive assessment of the swarm's ability to maintain safe separation, synchronized motion, and cohesive behavior. At some point in the simulation (after 60 seconds of simulation, considering agents maximum velocity), the swarm enters the equilibrium state, in which the collective motion has already stabilized, and agents have aggregated into their final swarm configuration. Only data from the equilibrium period relevant for assessing flocking is considered; to obtain this, we analyze only the last 25% of steps. Performance metrics are calculated for each agent at each step and then averaged. Then, run performance metrics are computed as the mean values of the corresponding step performance metrics. To obtain generalized performance metric values for a specific configuration (such as swarm size, swarm density level, neighbor selection strategy, and environment), multiple runs are conducted. Final performance metrics for a specific configuration are calculated as the mean values from these multiple runs.

The dual simulation environments (Section 4.5) – point mass dynamics for large swarms and Gazebo with PX4 for realistic quadcopter dynamics – enable a balanced evaluation. The point mass environment supports rapid, large-scale simulations to gather data, while Gazebo provides information about real-world applicability, incorporating aerodynamic and inertial effects. The lockstepping feature and MAVSDK-Python integration ensure synchronized, controlled experiments, mitigating computational problems as swarm size increases.

We design our experiments as follows: the swarm is assigned to perform goal-directed motion in a formation. To achieve this, each agent is given a constant migration velocity vector along the horizontal axis, regulated by migration gain to balance movement toward the goal with local flocking interactions. This parameter is critical for the experiments because large values could cause agents to move toward the goal independently rather than as a formation, leading to fragmentation. Conversely, if migration gain is too small, the swarm may prioritize local interactions and fail to perform the intended migration, stalling the mission. At the start of each run, agents are spawned randomly inside a cube, the size of which depends on the number of agents to ensure similar initial density across different configurations. We enforce a minimum inter-agent distance constraint in our spawn algorithm to prevent collisions from the start. Additionally, we include a rule that each agent must have at least one other agent within a specified radius to avoid initial fragmentation of the swarm. To ensure consistency across configurations, this radius is defined by a threshold distance used for the Metric neighbor selection strategy. However, in the Gazebo environment, some additional preparatory steps are necessary. In Gazebo, quadrocopter must reach the required thrust level to maintain height, so agents are initially spawned on the ground with subsequent thrust and takeoff commands. Once all agents reach their designated coordinates, the experiment begins. At this point, agents start receiving velocity commands generated by the flocking algorithm to assemble in formation while moving toward the goal, in both environments. Each run lasts the same amount of time (steps). When the run ends, performance metrics of the swarm can be collected.

Together, these components form a robust experimental framework to investigate vision-based localization in 3D environments. By simulating swarms of varying sizes (10 to 150 agents) and densities levels, we assess how occlusions degrade performance and how neighbor selection strategies mitigate these effects. The comparison between vision-based and communication-enabled swarms tests the necessity of communication, aiming to validate scalable, infrastructure-independent flocking.

5. Results of investigating vision-based UAV swarm flocking performance under visual occlusions in 3D environment

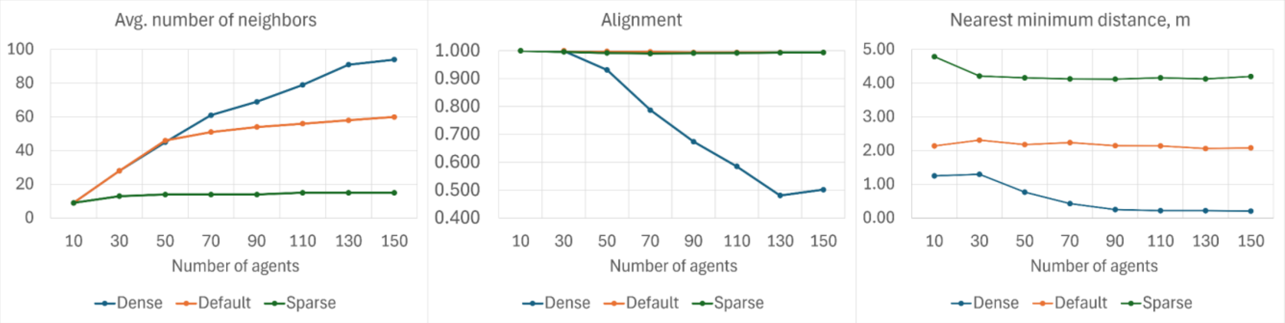
5.1. Results of enhancing swarm flocking method with neighbor selection strategy

We present experimental results evaluating the performance of vision-based UAV swarms in 3D environments, emphasizing the novel neighbor selection method developed to mitigate visual occlusions and achieve communication-free coordination. Our experiments assess swarm behavior across varying density levels, neighbor selection methods, and communication scenarios, using metrics such as minimum nearest neighbor distance (collision avoidance), alignment (velocity synchronization), and union (cohesion).

The reported metrics are mean values derived from the averaged data over 10 simulation runs per configuration, ensuring robust representation of swarm performance. The low variability across runs, as observed in the consistent metric trends, supports the reliability of these mean values, without evident long tails or high scatter that would undermine the conclusions. The agent radius is set to $r = 0.25\,\mathrm{m}$, reflecting typical quadcopter dimensions, and the perception radius is $r^{\max} = 10\,\mathrm{m}$, representing the range of onboard vision systems [5–9]. The maximum speed is $v^{\max} = 1\,\mathrm{m/s}$, consistent with common quadcopter capabilities. At the start of each simulation, agents are randomly spawned within a cube, with a minimum separation of $r^{\min} = 1\,\mathrm{m}$ and a maximum of $r^{\max}$, ensuring consistent density across swarm sizes. The migration velocity is set to $v^{mig} = [1, 0, 0]$ along the horizontal axis, with a migration gain of $k^{mig} = 0.5\,\mathrm{m/s}$, balancing goal-directed movement with formation stability. The time step is $\Delta t = 0.1\,\mathrm{s}$, sufficient for visual localization and control tasks, and the simulation duration is $T = 120\,\mathrm{s}$, allowing the swarm to achieve formation and migrate. Performance metrics are calculated for the last 25% of the simulation to focus on steady-state behavior. All metrics are calculated for every discrete time step k and then averaged.

The results are organized into two subsections, each addressing a specific objective: occlusion impact and neighbor selection efficiency and vision-based versus communication-enabled performance, with novel contributions highlighted through experimental data and figure analyses.

We developed a novel neighbor selection method (metric, topographic, and Delaunay strategies) to mitigate visual occlusions. We evaluated its impact on swarm performance across dense, default, and sparse configurations in a point mass dynamics environment. We tested three swarm densities levels – dense, default, and sparse – by tuning separation and cohesion gains. For dense swarms, we set $k^{sep}_{dense} = 1\,\mathrm{m/s}$ and $k^{coh}_{dense} = 3\,\mathrm{m/s}$; for default swarms $k^{sep}_{default} = 1\,\mathrm{m/s}$ and $k^{coh}_{default} = 1\,\mathrm{m/s}$; and for sparse swarms $k^{sep}_{sparse} = 5\,\mathrm{m/s}$ and $k^{coh}_{sparse} = 1\,\mathrm{m/s}$. Using the point mass dynamics environment, we simulated swarms of 10 to 150 agents $N \in \{10, 30, 50, 70, 90, 110, 130, 150\}$, relying solely on onboard vision-based localization for navigation. Performance was evaluated through minimum nearest neighbor distance (indicating collision avoidance), alignment (measuring velocity synchronization), and union (assessing swarm cohesion). Metrics are illustrated in Figure 1: Figure 1*a* shows the average number of neighbors detected by each agent, Figure 1b displays alignment values, and Figure 1*c* presents the minimum nearest neighbor distance across swarm sizes and density levels.



a *b* *c*

Fig. 1. Performance metrics for different swarm densities: *a* – Average number of neighbors; *b* – Alignment; *c* – Minimum nearest distance.

The results reveal that purely vision-based localization significantly impacts dense swarms compared to default and sparse ones. In dense formations, where agents are closely spaced, average number of neighbors increases significantly (Figure 1*a*). This causes an increased number of visual occlusions. This oversaturation of neighbors reduces localization accuracy, causing a marked decline in alignment (Figure 1*b*), with values dropping below 0.9 for swarms exceeding 70 agents. The minimum nearest neighbor distance also decreases significantly (Figure 1*c*), approaching critical thresholds (below 0.5 m) where collisions become imminent, particularly in dense swarms with more than 70 agents. Default and sparse swarms, with greater inter-agent spacing, exhibit less severe performance degradation, maintaining alignment above 0.9 and safer distances (above 0.5 m) across most swarm sizes. The union metric, consistently at 1 for all configurations, indicates that occlusions do not cause swarm fragmentation, as agents remain interconnected despite localization problems.

To address the performance degradation observed in dense swarms, we applied neighbor selection strategies (metric, topological, and Delaunay) to limit the number of agents considered as neighbors for flocking. It tends to reduce occlusion effects and enhance scalability. Experiments focused on dense swarms, which are most affected by occlusions, using the point mass dynamics environment for swarms of 10 to 150 agents $N \in \{10, 30, 50, 70, 90, 110, 130, 150\}$. To prevent initial disconnection in metric neighbor selection, all agents have at least one neighbor within $r^{metric} = 4\,\mathrm{m}$, the distance threshold for metric strategy. Parameters used for dense swarm introduced above are used throughout experiments. Results are presented in Figure 2: Figure 2*a* shows the average number of neighbors, Figure 2*b* displays alignment, Figure 2*c* illustrates minimum nearest neighbor distance, and Figure 2*d* depicts the union metric across swarm sizes for each strategy. Furthermore, we conducted experiments in Gazebo environment with realistic quadcopter dynamics for swarms up to 70 agents to verify the results $N \in \{10, 30, 50, 70\}$, as shown in Figure 3: Figure 3*a* (average neighbors), Figure 3*b* (alignment), Figure 3*c* (minimum distance), and Figure 3*d* (union).

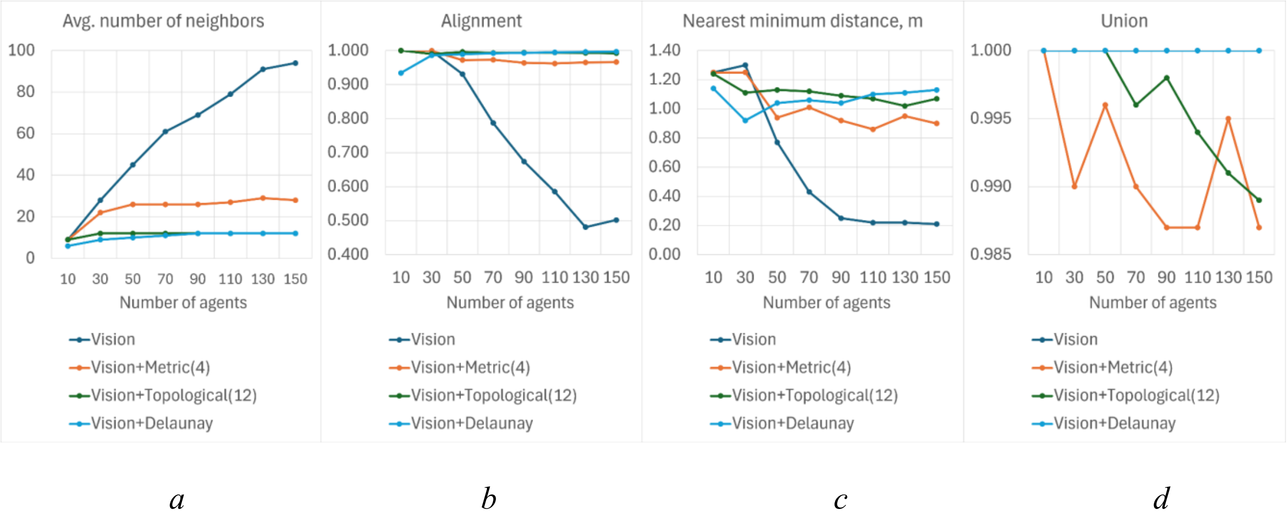


Fig. 2. Performance metrics for different neighbor selection strategies: *a* – Average number of neighbors; *b* – Alignment; *c* – Minimum nearest distance; *d* – Union.

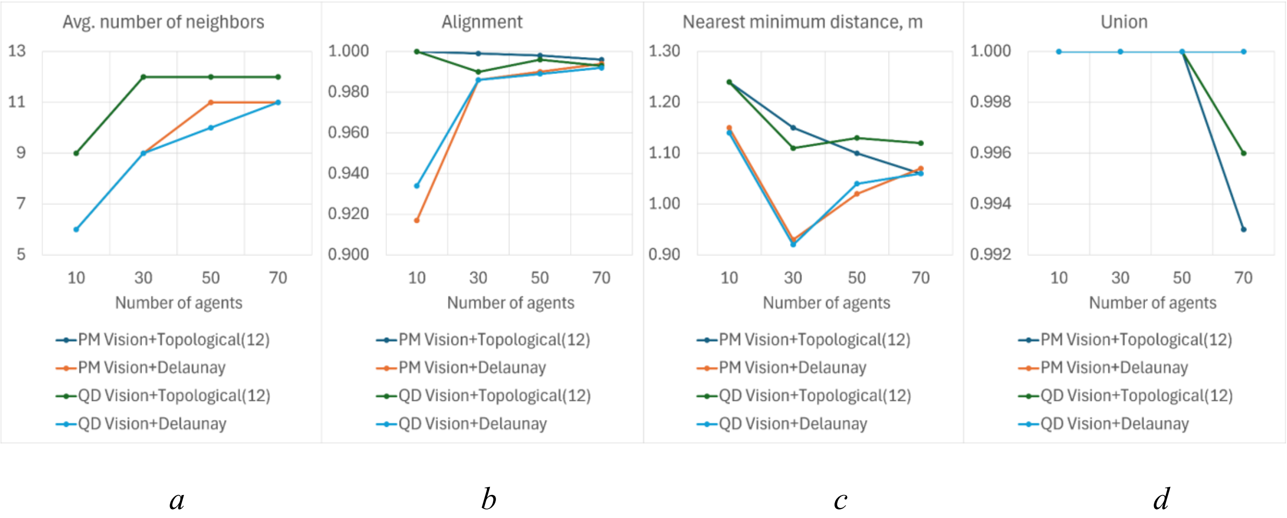


Fig. 3. Performance comparison of material point and quadrocopter dynamics: *a* – Average number of neighbors; *b* – Alignment; *c* – Minimum nearest distance; *d* – Union.

Implementing neighbor selection strategies significantly improves swarm performance. By restricting neighbor counts, all strategies maintain a semi-constant minimum distance (around 1 m, Figure 2*c*), effectively preventing collisions even as swarm size increases. Alignment improves notably, with topological and Delaunay strategies achieving values above 0.9 for most swarm sizes (Figure 2*b*), indicating synchronized motion. Metric selection, however, shows a gradual decline in alignment for larger swarms (above 90 agents), suggesting it is less robust for scalability. For smaller swarms (up to 30 agents), Delaunay selection exhibits slightly lower alignment (around 0.85) compared to topological (above 0.9), likely due to its spatially balanced but less consistent neighbor selection in sparse configurations. For larger swarms (above 30 agents), Delaunay matches or exceeds topological performance, maintaining high alignment (above 0.9).

The union metric (Figure 2*d*) reveals differences in cohesion. Metric and topological selections show fragmentation risks, with union values dropping below 1 for swarms of 30 (metric) and 50 (topological) agents, indicating subgroups forming due to inconsistent neighbor connections. Delaunay selection maintains perfect cohesion (union = 1) across all swarm sizes.

Comparison of point mass and Gazebo environments results shows similar trends (Figure 3). Topological selection maintains higher alignment (above 0.9) for swarms up to 50 agents but shows fragmentation at 50 agents (union < 1). Delaunay selection sustains viable alignment (around 0.9), which increases with the swarm size, and perfect cohesion (union = 1), reinforcing its suitability for larger swarms.

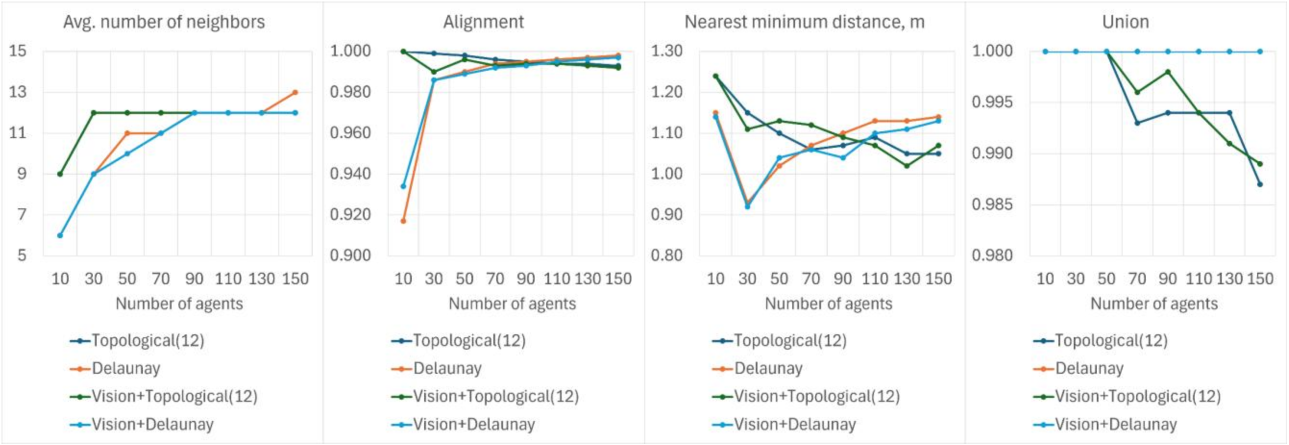
The presented metric values (Figures 1-3) represent mean values averaged over the 10 repeated simulation runs for each configuration, focusing on the last 25% of each run to capture steady-state behavior. The results showed consistent patterns across runs, with low variability observed, indicating reliability of the findings without notable long tails or high scatter in the metric distributions. This observed consistency aligns with the deterministic aspects of the simulations, such as fixed gains and migration velocity, though detailed statistical measures like standard deviation or skewness were not required given the stable outcomes.

The consistency between environments underscores the robustness of these strategies in realistic settings, where aerodynamic and inertial effects are considered. Neighbor selection strategies significantly enhance dense swarm performance by mitigating occlusion effects, maintaining safe distances, and improving alignment. Topological selection excels for smaller swarms (up to 50 agents), while Delaunay is superior for larger ones, ensuring cohesion and scalability. Metric selection, while effective for collision avoidance, is less robust for alignment and cohesion in larger swarms.

5.2. Results of performance comparison between vision-based swarm with applied neighbor selection strategy and communication-enabled swarm

We assessed our vision-based neighbor selection method against communication-enabled swarms to determine if communication-free coordination achieves comparable performance, using experimental data from point mass simulations of 10 to 150 agents. All agents in a communication-enabled swarm are aware of the positions of other agents. We selected topographic and Delaunay strategies, which outperformed metric selection in Section 5.1, for their occlusion-resilient properties. The results are presented in Figure 4: Figure 4*a* shows the average number of neighbors, Figure 4*b* displays alignment, Figure 4*c* illustrates minimum nearest neighbor distance, and Figure 4*d* depicts the union metric for vision-based and communication-enabled swarms.

Our contribution demonstrates that vision-based swarms, using our neighbor selection method, achieve performance comparable to or better than communication-enabled swarms, a significant advancement in decentralized coordination. Topographic selection yields higher alignment (above 0.9) for swarms up to 50 agents compared to communication-enabled swarms (around 0.85, Figure 4b), due to reduced occlusion noise. Delaunay selection maintains high alignment (above 0.9) and perfect cohesion (union = 1) for larger swarms, matching or surpassing communication-enabled performance (Figure 4*d*). Minimum distances remain comparable (around 1m, Figure 4*c*), ensuring equivalent collision avoidance. Topographic selection shows fragmentation risks above 50 agents (union < 1, Figure 4*d*), while Delaunay's robust cohesion makes it ideal for scalability.



a *b* *c* *d*

Fig. 4. Performance metrics for vision-based neighbor selection strategies in comparison with all-to-all connected strategies: *a* – Average number of neighbors; *b* – Alignment; *c* – Minimum nearest distance; *d* – Union.

This result validates the feasibility of communication-free, occlusion-resilient flocking in 3D environments, advancing swarm robotics by eliminating external dependencies, unlike prior studies reliant on communication.

6. Discussion of results regarding vision-based swarm flocking under visual occlusions in 3D environment

This study advances swarm robotics by developing a neighbor selection method for occlusion-resilient, vision-based UAV swarm coordination in 3D environments, addressing the problem of visual occlusions. Our results demonstrate that occlusions from neighboring agents impair dense swarm performance, driven by oversaturated neighbor detection that disrupts vision-based localization. Sparse and default configurations experience milder setbacks due to fewer occlusions, enabling stable alignment (above 0.9) and safer distances (above 0.5m). The observed degradation in dense swarms – alignment dropping below 0.9 and distances below 0.5m for swarms exceeding 70 agents – stems from increased occlusion frequency, which reduces localization accuracy and increases collision risks. This analysis highlights the need for density management and occlusion-mitigating strategies to ensure collision-free flocking.

Our neighbor selection method, comprising metric, topographic, and Delaunay strategies, effectively mitigates occlusion effects representing a novel contribution to swarm robotics. All strategies maintain consistent inter-agent distances (around 1m, Figure 2*c*), minimizing collision risks. Topographic selection excels in small swarms (up to 50 agents) because its fixed neighbor count reduces occlusion noise, achieving superior alignment (above 0.9, Figure 2*b*), consistent with prior findings on limited interactions [9–11, 14]. However, its fragmentation in larger swarms (union < 1 above 50 agents, Figure 2*d*) results from inconsistent neighbor connections. Conversely, Delaunay selection's spatially balanced tetrahedral mesh ensures perfect cohesion (union = 1) across all swarm sizes and high alignment (above 0.9) for larger swarms. However, it shows slightly lower alignment (around 0.85) in small swarms due to sparse connections (Figures 2*b*, 2*d*). Metric selection, less effective, struggles with cohesion (union < 1 above 30 agents) and alignment due to variable neighbor counts. These results, validated in Gazebo simulations (Figure 3), confirm that topographic selection is optimal for smaller swarms, while Delaunay excels for larger ones, offering a scalable solution for large-scale missions such as environmental monitoring. Our results align with prior works on vision-based swarm challenges but extend them to 3D environments. For instance, it was reported occlusion-induced performance degradation in 2D settings [9], with similar declines in alignment and cohesion, though without quantitative statistical measures like standard deviation in their analysis. Our neighbor selection method achieves higher alignment (above 0.9) and perfect cohesion in 3D, surpassing their findings by addressing spatial occlusions, which were not explored in their planar focus. The observed consistency in our mean values across runs reinforces the robustness of these personal contributions, distinguishing our work from existing literature that often overlooks 3D dynamics or reports results without scatter analysis.

Our comparison of vision-based and communication-enabled swarms reveals that our neighbor selection method enables communication-free coordination that matches or surpasses communication-enabled performance. Topographic selection achieves higher alignment (above 0.9 vs. 0.85, Figure 4*b*) in small swarms by minimizing occlusion noise, while Delaunay maintains high alignment and perfect cohesion in larger swarms (Figures 4*b*, 4*d*), driven by its robust neighbor connections. Comparable minimum distances (around 1m, Figure 4*c*) indicate equivalent collision avoidance. This novel finding eliminates the need for communication, freeing bandwidth for data transmission and enhancing resilience in GNSS-denied environments. In conclusion, our method addresses the occlusion problem by enabling scalable flocking with topographic and Delaunay strategies, optimizing performance for various swarm sizes, and advancing decentralized swarm robotics.

Despite these advances, our occlusion model assumes a spherical agent shape, which might not accurately reflect the complex geometry of UAVs. However, since UAVs can vary in shape, carry extra equipment such as landing gear, or have different payloads attached to the top or bottom, a spherical model remains a practical alternative. It provides a more general approach to modeling UAV geometry for the occlusion model, while enabling large-scale simulations by reducing computational complexity. We believe that this model strikes a balance between realism and efficiency, while accounting for the complex shapes of real-world UAVs. Future research may focus on refining the model to enhance the practicality of our research for real-world applications and deployments. Exploring reinforcement learning (RL) [13] could further enhance occlusion mitigation and scalability, while real-world experiments would validate our findings under dynamic conditions. RL can be used to develop a new neighbor selection strategy that does not depend on swarm size. Alternatively, RL algorithms can improve flocking algorithms, making them more suitable for real-world environments with many obstacles, which could also affect the swarm performance. These prospects promise to advance swarm robotics for applications addressing the growing demand for autonomous, infrastructure-independent systems.

The research was implemented within the National Research Foundation of Ukraine project No. 2023.04/0077 “Drone for water sampling”.

Conclusions

Our study establishes the feasibility of occlusion-resilient, vision-based UAV swarm coordination in 3D environments. We developed a novel vision-based UAV swarm flocking method suitable for 3D environments, an area that has been under-researched in previous works. This method utilizes neighbor selection strategies such as metric (considering only agents within the threshold), topological (considering only n-nearest agents), and Delaunay (considering agents that share ridges produced by Delaunay triangulation). Neighbor selection restricts neighbor count, thus mitigating visual occlusions. This discovery enables significant advancements in swarm performance. Experimental data reveal that, without neighbor selection, dense swarms suffer degraded alignment (below 0.9) and reduced minimum distances (below 0.5 m) for swarms exceeding 70 agents, increasing collision risks. By applying proposed selection strategies, we can boost swarm performance to achieve levels adequate for real-world deployments. For instance, topographic selection ensures high alignment (above 0.9) and stable distances (around 1 m) in small swarms (up to 50 agents). In contrast, Delaunay selection maintains perfect cohesion (union = 1) and high alignment across all swarm sizes, especially excelling for larger swarms. This novel method enhances scalability and safety in dense 3D formations, offering practical value for applications where occlusion-prone environments are common.

Further experiments comparing purely vision-based swarms with communication-enabled ones reveal that the optimal vision-based neighbor selection method enables communication-free swarm coordination that matches or exceeds the performance of communication-enabled swarms.

Topographic selection achieves superior alignment (above 0.9 vs. 0.85) in small swarms, while Delaunay selection sustains high alignment and cohesion in larger swarms, with comparable collision avoidance (distances around 1 m). This eliminates the need for inter-agent communication, providing a robust, infrastructure-independent alternative that enhances operational efficiency in GNSS-denied settings, such as disaster response or urban surveillance. These conclusions, validated in Gazebo simulations, underscore the practical significance of our method for scalable, autonomous UAV swarm deployments, thereby advancing prior work that relies on communication or simplified 2D models.

References

[1] M. Abdelkader, S. Güler, H. Jaleel, and J. S. Shamma, “Aerial swarms: recent applications and challenges,” *Current Robotics Reports*, vol. 2, no. 3, pp. 309–320, 2021, https://doi.org/10.1007/s43154-021-00063-4.

[2] D. Floreano and R. J. Wood, “Science, technology and the future of small autonomous drones,” *Nature*, vol. 521, no. 7553, pp. 460–466, 2015, https://doi.org/10.1038/nature14542.

[3] T. Cieslewski, S. Choudhary and D. Scaramuzza, "Data-Efficient Decentralized Visual SLAM," *IEEE International Conference on Robotics and Automation (ICRA)*, Brisbane, QLD, Australia, 2018, pp. 2466–2473, https://doi.org/10.1109/ICRA.2018.8461155.

[4] G. Vasarhelyi, C. Virágh, G. Somorjai, N. Tarcai, T. Szörenyi, T. Nepusz and T. Vicsek, “Outdoor flocking and formation flight with autonomous aerial robots,” *IEEE/RSJ International Conference on Intelligent Robots and Systems*, IL, USA, 2014, pp. 3866–3873, 2014, https://doi.org/10.1109/IROS.2014.6943105.

[5] X. Yan, H. Deng, and Q. Quan, “Active infrared coded target design and pose estimation for multiple objects,” *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Macau, China, 2019, https://doi.org/10.1109/IROS40897.2019.8967660.

[6] M. Petrlik, T. Baca, D. Hert, M. Vrba, T. Krajnik, and M. Saska, “A robust UAV system for operations in a constrained environment,” *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2169–2176, 2020, https://doi.org/10.1109/LRA.2020.2970980.

[7] P. Petráček, V. Walter, T. Báča, and M. Saska, “Bio-inspired compact swarms of unmanned aerial vehicles without communication and external localization,” *Bioinspiration & Biomimetics*, vol. 16, no. 2, p. 026009, 2020, https://doi.org/10.1088/1748-3190/abc6b3.

[8] F. Schilling, F. Schiano, and D. Floreano, “Vision-Based drone flocking in outdoor environments,” *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 2954–2961, 2021, https://doi.org/10.1109/LRA.2021.3062298.

[9] F. Schilling, E. Soria, and D. Floreano, “On the Scalability of Vision-Based Drone Swarms in the Presence of Occlusions,” *IEEE Access*, vol. 10, pp. 28133–28146, 2022, https://doi.org/10.1109/ACCESS.2022.3158758.

[10] R. Martínez-Clark, J. Pliego-Jimenez, J. F. Flores-Resendiz, and D. Avilés-Velázquez, “Optimum k-Nearest Neighbors for Heading Synchronization on a Swarm of UAVs under a Time-Evolving Communication Network,” *Entropy*, vol. 25, no. 6, p. 853, 2023, https://doi.org/10.3390/e25060853.

[11] V. Kumar and R. De, “Efficient flocking: metric versus topological interactions,” *Royal Society Open Science*, vol. 8, no. 9, 2021, https://doi.org/10.1098/rsos.202158.

[12] C. W. Reynolds, “Flocks, herds and schools: A distributed behavioral model,” *ACM SIGGRAPH Computer Graphics*, vol. 21, no. 4, pp. 25–34, 1987, https://doi.org/10.1145/37402.37406.

[13] Y. Albrekht and A. Pysarenko, “Exploring the power of heterogeneous UAV swarms through reinforcement learning,” *Technology Audit and Production Reserves*, vol. 6, no. 2(74), pp. 6–10, 2023, https://doi.org/10.15587/2706-5448.2023.293063.

[14] O. Smovzhenko, A. Pysarenko, " Evaluating Vision-Based Drone Swarms Performance under Visual Occlusions", *13th International Scientific And Practical Conference On Information Systems and Technologies Infocom Advanced Solutions*, Kyiv, Ukraine, 2025, pp. 42–44. [Online]. Available: https://ist.kpi.ua/wp-content/uploads/2025/05/infocom-advanced-solutions-2025_compressed-3.pdf.

УДК 004.8

МЕТОД ВИБОРУ СУСІДІВ НА ОСНОВІ ВІЗУАЛЬНОЇ ІНФОРМАЦІЇ ДЛЯ КООРДИНАЦІЇ РОЮ БЕЗПІЛОТНИХ ЛІТАЛЬНИХ АПАРАТІВ СТІЙКИХ ДО ВІЗУАЛЬНИХ ПЕРЕШКОД У ТРИВИМІРНИХ СЕРЕДОВИЩАХ

Олексій Смовженко
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна
https://orcid.org/0009-0006-0500-5005

Андрій Писаренко
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна
https://orcid.org/0000-0001-7947-218X

Рої безпілотних літальних апаратів (БПЛА) забезпечують більшу масштабованість, надійність та ефективність у порівнянні з використанням окремих БПЛА. Ці переваги дозволяють використовувати рої БПЛА для пошуково-рятувальних операцій, у сільському господарстві, екологічному моніторингу та спостереженні у міському середовищі. З іншого боку, такі рої вразливі до втрати сигналу, глушіння та обмеженої масштабованості через залежність від глобальних навігаційних супутникових систем та бездротового зв’язку. Це дослідження сприяє розвитку роєвої робототехніки за рахунок створення нового стійкого до візуальних перешкод методу вибору сусідів для дронів на основі візуальних даних в тривимірному середовищі (3D). Цей метод дозволяє вирішити проблему, спричинену візуальними перешкодами, яка здатна порушити децентралізовану координацію рою. Ми розглядаємо задачу координації рою у вигляді штучного потенційного поля, на відміну від попередніх досліджень, які фокусувались на двовимірних середовищах або ж на використанні зв’язку. Для оцінки продуктивності використовуються такі метрики, як мінімальна відстань до найближчого сусіда (уникнення зіткнень), узгодженість (синхронізація швидкостей) та єдність. Використання середовища з динамікою матеріальної точки та середовища з більш реалістичною динамікою квадрокоптера дозволяє нам оцінити поведінку рою у випадку тісної, звичайної та розосередженої конфігурації. Результати, отримані нами, показують, що перешкоди негативно впливають на узгодженість (нижче 0.9) та мінімальну відстань (менше 0.5м) у тісному рою, який налічує більше 70 агентів, що підвищує ймовірність зіткнень. Запропонований нами метод, який використовує метричну, топографічну стратегії або ж стратегію з використанням тріангуляції Делона, дозволяє уникнути цих проблем. Топографічний вибір дозволяє досягти високого рівня узгодженості (більше 0.9) для невеликих роїв (до 50 агентів). З іншого боку використання тріангуляції Делона забезпечує ідеальну єдність та високу узгодженість для роїв всіх розмірів. Дані твердження підкріплені результатами, отриманими за допомогою симуляцій. Наш метод дозволяє позбавитись використання зв’язку для координації рою, забезпечуючи або перевищуючи продуктивність рою зі зв’язком. Це можливо за рахунок використання топологічного вибору сусідів (узгодженість 0.9 проти 0.85) для невеликих роїв та вибору на основі тріангуляції Делона для більших. Наведені результати доводять, що використання зв’язку між агентами не є необхідним для координації рою. Відповідно, це дозволяє підвищити стійкість рою та пропускну здатність каналів зв’язку для передачі іншої інформації. Використання запропонованого методу дозволяє створити масштабований, незалежний від інфраструктури фреймворк для роїв БПЛА, що несе практичну цінність для автономних операцій роїв в складних середовищах з високою кількістю візуальних перешкод.

Ключові слова: рій БПЛА, візуальна локалізація, контроль формації, децентралізована координація, штучне потенціальне поле.