Efficient UAV Coverage Algorithms for Areas Requiring Heterogeneous Sensors

TARAVAT DARBOUEI
Dept. of Computer Science
Kookmin University
Seoul, South Korea
taravatdarbouei@kookmin.ac.kr

SANGHWAN LEE
Dept. of Computer Science
Kookmin University
Seoul, South Korea
sanghwan@kookmin.ac.kr

***Abstract*—Unmanned Aerial Vehicles (UAVs) are increasingly used for line coverage, mapping, and surveillance, where efficiency and cost-effectiveness are critical. A key challenge in real-world deployments is that required edges often demand different sensors, yet most prior studies assume UAVs carry only a single sensor type. This work addresses UAV coverage optimization under strict time and energy limits in heterogeneous sensor environments. We develop and compare two strategies: (i) a Greedy algorithm, where each UAV carries a single sensor and services only compatible edges, and (ii) a Multiple Sensors algorithm, where each UAV carries all required sensors and can cover any edge. Using urban datasets from New York, London, and Tehran, we demonstrate distinct trade-offs. The Greedy algorithm achieves the lowest overall operational cost by balancing fleet size and energy consumption, whereas the Multiple Sensors algorithm minimizes the number of UAVs required but incurs higher per-UAV energy usage due to increased payload. These findings provide practical guidelines for selecting deployment strategies based on mission priorities—whether minimizing fleet size or reducing total cost—and support UAV applications in urban planning, disaster response, and smart city operations.**

***Index Terms*—UAVs, Sensors, Line coverage, Multi-sensor UAVs, Heterogeneous environments, Coverage optimization**

I. INTRODUCTION

Unmanned Aerial Vehicles (UAVs) have emerged as indispensable tools in disaster response, infrastructure inspection, surveillance, and environmental monitoring [1]. Their ability to access hazardous or remote areas enables rapid assessment of conditions—such as flood-damaged roads or malfunctioning power lines—supporting faster and more informed decision-making. However, inherent limitations in battery capacity, payload weight, and flight endurance necessitate careful operational planning to ensure complete and efficient coverage of target areas [1].

To address these challenges, researchers have investigated various coverage strategies tailored to different operational needs. Among these, a key paradigm is *line coverage*, where UAVs traverse and inspect linear infrastructures (e.g., roads, pipelines, transmission lines) under strict time and constraints [2], [3]. In *sensor-diverse environments*, each required edge may demand a specific sensor (RGB camera, LiDAR, infrared, radar, ultrasonic), which complicates both resource allocation and routing [4]. Achieving efficiency in such settings involves addressing:

- **Optimal Path Planning:** Minimizing travel distance and energy consumption while ensuring full edge coverage [5]–[7].
- **Sensor Allocation:** Matching UAV sensing capabilities to edge requirements; evaluating trade-offs between single-sensor platforms and multi-sensor payloads [4], [8].
- **Coordination:** Managing multiple UAVs operating over large or obstructed areas [9], [10].
- **Battery/Energy Management:** Balancing energy use between active coverage and deadheading; selecting speeds and routes that respect endurance limits [5], [6].

Traditional routing approaches often assume homogeneous sensors and can underperform when sensing requirements vary across edges [2], [3]. This paper examines two strategies tailored for heterogeneous environments:

1) **Greedy with diverse sensors** — each UAV carries a single sensor type and services the nearest compatible required edge.
2) **UAVs with multiple sensors** — each UAV carries all required sensors, enabling unrestricted coverage at the cost of higher payload and energy consumption.

We evaluate both algorithms on five real-world urban datasets—New York, London, Paris, Istanbul, and Tehran—under identical operational constraints. The Greedy approach is expected to reduce total operational cost, while the Multiple Sensors approach aims to minimize fleet size.

The contribution of this paper is summarized as follows:

- Development and implementation of two UAV coverage algorithms designed for heterogeneous sensor requirements.
- Comparative evaluation across multiple cities, analyzing UAV count, energy consumption, and operational cost.
- Practical guidance on when to prioritize cost efficiency (Greedy) versus minimizing fleet size (Multiple Sensors) in real deployments.

The rest of the paper is organized as follows. Section II reviews related works and background on UAV coverage, sensor allocation, and energy-aware planning. Section III formally defines the problem and presents the modeling assumptions. Section IV details the two proposed algorithms—Greedy with diverse sensors and UAVs with multiple sensors. Section V reports simulation results and comparative analysis across multiple city datasets. Finally, Section VI concludes the paper and discusses directions for future research.

II. Related Works and Background

UAV-based line coverage has been investigated for mapping, inspection, and surveillance [1]–[3]. Core challenges include efficient route planning, sensor allocation, and energy optimization, particularly when each edge requires a specific sensing modality [4]. Foundational work on line coverage with one or more robots provides theoretical and experimental baselines for traversing required edges efficiently [2], [3], [11]. In practice, minimizing both servicing and deadhead motion under endurance and kinematic constraints is crucial for scalability to realistic road networks [2], [3], [12]. Multi-UAV sweep-coverage strategies further explore coordination and task partitioning to reduce mission time [9], [10]. Sensor selection strongly influences mission performance and feasibility. RGB cameras support high-resolution imaging; LiDAR enables 3D mapping; infrared sensors capture thermal signatures; radar/ultrasonic systems assist in obstacle detection and safety [4]. Multi-sensor fusion frameworks improve robustness and accuracy but increase payload mass and energy consumption [8]. These trade-offs motivate our comparison of (i) single-sensor Greedy assignment versus (ii) multi-sensor UAVs that can service any edge. Energy-aware coverage emphasizes reducing deadhead distance and planning smoother, kinematically feasible trajectories. Online coverage planning that accounts for energy [5], surface/graph shortest paths (e.g., extended Dijkstra) [6], and occlusion-aware reconnaissance in complex environments [7] have all demonstrated measurable energy savings and improved coverage consistency. These principles inform our evaluation setup (speeds, power models, and return-to-depot constraints). While prior studies separately address route optimization [2], [3], [5]–[7], [12] and sensor considerations [4], [8], fewer works evaluate their combined impact in heterogeneous settings. We address this by comparing two sensor-allocation strategies under identical routing and operational assumptions, reporting on energy use, UAV count, and cost.

III. Problem Statement

This study addresses the UAV coverage problem in a heterogeneous sensor environment, where each required edge in a network demands a specific sensing capability (e.g., RGB, LiDAR, infrared). The goal is to cover all required edges efficiently while minimizing total energy consumption and operation time, under a strict mission time limit.

UAV operations consist of:

- **Servicing:** Traversing required edges with the appropriate active sensor, consuming higher energy at the service speed S_{svc}.
- **Deadheading:** Traveling along non-required edges (logistical movement) without active sensing, consuming less energy at a higher deadhead speed S_{dh}.

Each UAV operates under a maximum mission duration T_{lim}, and is dynamically allocated based on available time, energy, and sensor compatibility. A UAV is deactivated once it cannot service the next required edge, and a new UAV is deployed. This problem can be modeled as a variant of the Vehicle Routing Problem (VRP) with constraints on time, energy, and sensor assignment.

We define the notations used throughout this paper as follows:

- $Lat(u), Lon(u)$: Latitude and longitude of vertex u.
- $d(u, v)$: Euclidean distance between two vertices u and v.
- N: Total number of UAVs used.
- T_{lim}: Maximum operating time per UAV.
- P_s, P_{dh}: Power consumption in service and deadhead modes (W).
- S_{svc}, S_{dh}: Service and deadhead speeds (m/s).
- $Sensor(e)$: Sensor type required to service edge e.
- $Sensors(u)$: Set of sensor types available on UAV u.
- P_{depot}: Selected starting depot for each UAV (chosen based on shortest distance to required edges).
- $P(u, v)$: Shortest path between vertices u and v.
- T_s, T_d: Time required for service mode and deadhead traversal, respectively.
- E_s, E_d: Energy consumed during servicing and deadhead traversal, respectively.

The time and energy are modeled as follows. Service time and energy for a required edge of length d:

$$T_s = \frac{d}{S_{svc}}, \quad E_s = \frac{P_s \cdot d}{3600 \cdot S_{svc}}$$

Deadhead time and energy:

$$T_d = \frac{d}{S_{dh}}, \quad E_d = \frac{P_{dh} \cdot d}{3600 \cdot S_{dh}}$$

Total energy per UAV:

$$E_{total} = \sum_{e \in E_{req}} E_s(e) \; + \sum_{e \in E_{nonreq}} E_d(e)$$

IV. Algorithms

This work evaluates two UAV coverage algorithms addressing the line coverage problem in sensor-diverse environments: the *Greedy Algorithm with Diverse Sensors* and the *UAVs with Multiple Sensors* algorithm. Both approaches aim to maximize coverage efficiency and minimize operational costs, but differ significantly in how sensor assignments influence UAV path planning.

Focus of this Study. Two coverage strategies are compared:

1) **Greedy with diverse sensors:** Each UAV carries one sensor type, assigned to edges it can service.
2) **Multiple Sensors:** Each UAV carries all required sensors, enabling it to cover all edges without reassignment, potentially reducing fleet size.

For depot selection in both algorithms, the UAVs choose the closest depot using Dijkstra's algorithm to compute the shortest path between each depot and the nearest required edge. The process considers both required and non-required edges as potential travel paths. After evaluating all depots, the UAV departs from the one with the minimum travel distance, thereby reducing unnecessary deadheading and optimizing energy usage.

Greedy Algorithm Explanation with Example:

In the Greedy algorithm, they service edges based on proximity while checking if they have the correct sensor for each edge. The random assignment of sensors to the required edges still applies here, meaning each UAV can only service an edge if it has the corresponding sensor. After departing from the depot, each UAV dynamically chooses the closest required edge. However, it will only service an edge if it has the correct sensor type. For example, if the nearest edge requires a camera sensor, the UAV must have a camera sensor to service it. If the UAV does not have the necessary sensor, it will continue to search for the closest required edge it can service or deadhead to another edge.

The UAVs continue servicing edges based on proximity and sensor capability until they either run out of time limit or all required edges are serviced. The non-required edges are still used for deadheading between required edges. Once a UAV runs out of time or finishes its assigned edges, it returns to the depot. In this algorithm, the UAVs have more freedom to choose edges, but they must still meet the sensor requirements for each edge.

In Figure 1, the Greedy algorithm, UAVs cover required edges based on sensor type by dynamically choosing the closest serviceable edge. For example, with 16 edges (7 required), edges 6–10 may need a camera sensor while edges 5–6 require a LiDAR sensor. The UAVs start from the nearest depot, which is depot 10 in this case. A UAV with a camera sensor will first cover edge 10–6. After servicing the edge, it checks its remaining time: if sufficient, it proceeds to the next closest required edge it can service; otherwise, it returns to the depot using a deadhead path (non-required edges), such as $6 \rightarrow 7 \rightarrow 11 \rightarrow 10$. UAVs move freely across the map, covering the nearest required edge that matches their sensor capability. Dijkstra's algorithm determines the shortest path to each required edge, and UAVs continue until the time constraint is reached.

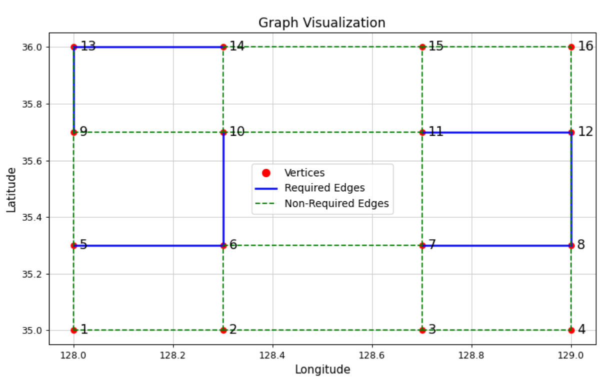

Fig. 1: Map 4 × 4 - Greedy

A. Greedy Algorithm with Diverse Sensors

In the Greedy algorithm, each UAV dynamically selects the next closest required edge based on proximity, provided it has the correct sensor for that edge. Sensors are assigned randomly to required edges at the start of the mission. A UAV can only service an edge if it possesses the corresponding sensor type. For example, if the nearest edge requires a LiDAR sensor but the UAV is equipped with a thermal sensor, it will skip that edge and search for the next closest edge it can service. The UAV continues this selection process until it reaches its maximum operational time limit, after which it returns to the depot.

Non-required edges are occasionally used for deadheading to reach the next required edge, ensuring operational continuity. This approach is simple and flexible, but may lead to inefficient coverage if sensor assignments limit UAV access to nearby edges, causing additional deadheading and unbalanced workload distribution among UAVs.

Algorithm 1 UAV Coverage Greedy Algorithm

Input: $\mathbf{V}, \mathbf{G}, \mathbf{P_{depot}}, \mathbf{T_{lim}}, \mathbf{S_{svc}}, \mathbf{S_{dh}}, \mathbf{E_{svc}}, \mathbf{E_{dh}}$
Output: $\mathbf{N}_{\text{UAVs}}, \mathbf{E}_{\text{covered}}$

Definitions:
V: Set of vertices in the graph
G: Graph structure with required and non-required edges
P_{depot}: Starting depot for all UAVs
N_{UAVs}: Total number of UAVs used
E_{covered}: Set of all covered edges

Initialize $E_{\text{covered}} \leftarrow \emptyset$, $N_{\text{UAVs}} \leftarrow 0$
Group required edges by sensor type: $E_{\text{sensor}} \leftarrow$ GROUPEDGESBYSENSOR(G)
for all $S_{\text{type}}, E \in E_{\text{sensor}}$ **do**
  while E contains uncovered edges **do**
    Create a new UAV u for S_{type} with starting position P_{depot}
    $N_{\text{UAVs}} \leftarrow N_{\text{UAVs}} + 1$
    while u is active and T_{lim} is not exceeded **do**
      Determine the closest uncovered edge $e_{\text{closest}} \in E$
      if $e_{\text{closest}} = \emptyset$ **or** u cannot return to P_{depot} after servicing e_{closest} **then**
        Break
      end if
      if u is not at the start of e_{closest} **then**
        Perform deadhead traversal to move UAV to e_{closest}.start
        Update UAV's time, energy, and position for deadhead traversal
      end if
      Service e_{closest}: Add e_{closest} to E_{covered}
      Remove e_{closest} from E
      Update UAV's position, time, and energy for servicing e_{closest}
      if UAV runs out of time or energy **then**
        Mark u as inactive and **Break**
      end if
    end while
    Return u to P_{depot} using **deadhead traversal** if not already at depot
  end while
end for
if any required edges remain uncovered **then**
  for all remaining uncovered edges $e \in G_{\text{req}}$ **do**
    Create a new UAV to cover e
    Add e to E_{covered}
    $N_{\text{UAVs}} \leftarrow N_{\text{UAVs}} + 1$
  end for
end if
return $N_{\text{UAVs}}, E_{\text{covered}}$

B. UAVs with Multiple Sensors

In the UAVs with Multiple Sensors algorithm, each UAV is equipped with all required sensor types, removing the constraints present in the Greedy approach. This capability allows UAVs to service any required edge regardless of the sensor type needed, eliminating inefficiencies caused by sensor mismatches. After departing from the closest depot (determined by Dijkstra's algorithm), each UAV services the nearest required edge based purely on proximity and remaining operational time. Since there are no sensor restrictions, UAVs do not need to bypass edges, reducing deadheading and improving overall mission efficiency. Non-required edges may still be used strategically for travel between required edges. UAVs operate until they reach the maximum time limit, ensuring full utilization of available operational capacity before returning to the depot. While this approach increases energy consumption per UAV due to the heavier payload of multiple sensors, it significantly reduces the total number of UAVs required, balances workload distribution, and shortens mission completion time. It is especially effective in scenarios with varied sensor requirements and dispersed coverage needs.

Algorithm 2 UAVs with Multiple Sensors Algorithm

Input: $\mathbf{V, G, D, T_{lim}, S_{svc}, S_{dh}, E_{svc}, E_{dh}}$
Output: $\mathrm{N}_{\mathrm{UAVs}}, \mathrm{U}_{\mathrm{details}}$
  Definitions:
    V: Set of vertices in the graph
    G: Graph with required and non-required edges
    D: Starting depot for all UAVs
    N_{UAVs}: Total UAVs used
    E_{req}: Set of required edges to cover
    U_{details}: UAV operations (paths, energy, time)
  Initialize $E_{\mathrm{covered}} \leftarrow \emptyset$, $N_{\mathrm{UAVs}} \leftarrow 0$
  while $E_{\mathrm{req}} \neq \emptyset$ **do**
    Create new UAV u with all sensor types
    Add u to UAV list; $N_{\mathrm{UAVs}} \leftarrow N_{\mathrm{UAVs}} + 1$
    while u is active **and** $E_{\mathrm{req}} \neq \emptyset$ **do**
      $E_{\mathrm{closest}} \leftarrow \texttt{None}$; $d_{\mathrm{min}} \leftarrow \infty$
      for all $e \in E_{\mathrm{req}}$ **do**
        Compute distance d from UAV to e_{start}
        if $d < d_{\mathrm{min}}$ **then**
          $E_{\mathrm{closest}} \leftarrow e$, $d_{\mathrm{min}} \leftarrow d$
        end if
      end for
      if $E_{\mathrm{closest}} = \texttt{None}$ **or** UAV cannot return to D after servicing E_{closest} **then**
        Mark UAV u as inactive and **break**
      end if
      if UAV is not at E_{closest}.start **then**
        Perform deadhead traversal to E_{closest}.start
        Update UAV's position, time, and energy **in** $U_{\mathbf{details}}$
      end if
      Cover E_{closest}:
      Add E_{closest} to $E_{\mathrm{covered}}, E_{\mathrm{covered\_req}}$
      Remove E_{closest} from E_{req}
      Update UAV's position, time, and energy **in** $U_{\mathbf{details}}$
    end while
    if UAV has remaining time and energy **then**
      Return UAV to D via **deadhead traversal**
      Update UAV's position, time, and energy **in** $U_{\mathbf{details}}$
    end if
  end while
  return $N_{\mathrm{UAVs}}, U_{\mathrm{details}}$

V. Evaluation

In this section, we evaluate the performance of the two proposed algorithms—Greedy and UAV with multiple sensors—by analyzing their efficiency, energy consumption, and coverage capabilities in line coverage tasks. Both algorithms were rigorously tested under time constraints, ensuring complete coverage of required edges across various urban environments.

The Greedy approach provided flexible edge traversal strategies to maximize coverage within the UAVs' operational limits. In contrast, the multiple-sensors algorithm demonstrated versatility by equipping UAVs with diverse sensor capabilities, enabling multi-purpose tasks and reducing the number of UAVs needed.

Each algorithm was tested using real-world city datasets, including New York and other representative cities, to provide diverse and comprehensive insights. The evaluation considered key performance metrics such as UAV count, energy consumption, coverage efficiency, and cost-effectiveness.

A. Methodology

The methodology focuses on comparing the performance of the Greedy algorithm and the UAV with multiple sensors algorithm in covering required edges within urban road networks. Both algorithms were evaluated using key performance metrics such as edge coverage, energy consumption, and UAV count. Simulations were designed under realistic conditions, where UAVs began from a depot, covered assigned edges, and returned to the depot before their time limits were depleted.

The **Greedy algorithm** prioritizes covering edges based on immediate cost-effectiveness, aiming to maximize coverage with minimal UAV deployment. The **UAV with multiple sensors algorithm**, in contrast, equips UAVs with diverse sensor capabilities (RGB, radar, infrared, LiDAR, ultrasonic), enabling a single UAV to perform multiple sensing tasks simultaneously, thus reducing the overall UAV count.

B. Line Coverage Dataset

The Line Coverage dataset, developed by the University of North Carolina at Charlotte's Robotics Group, is a collection of urban road networks derived from OpenStreetMap and released under the Open Database License (ODbL). It is designed to support research on line coverage problems, especially in applications involving Unmanned Aerial Vehicles (UAVs). The dataset captures the structural diversity of real cities, making it valuable for evaluating algorithm performance under realistic conditions.

The table represents key metrics for the urban road networks of five cities, Istanbul, Paris, New York, London, and Tehran, extracted from the Line Coverage dataset. These cities were chosen to provide diverse network structures and complexity levels, enabling comprehensive evaluation of UAV deployment strategies in real-world scenarios. The metrics include the number of nodes, required edges, non-required edges, total network length, and the number of connected components for each city, offering insights into the characteristics of their respective road networks.

These statistics highlight that Paris and Istanbul exhibit the densest and most complex road networks due to their high number of nodes and edges. In contrast, London shows the smallest network length, suggesting a more compact structure.

TABLE I: City's Information

City	No. of Nodes	No. of Required Edges	No. of Non-Required Edges	Length of Network (m)	No. of Connected Components
Istanbul	430	543	92235	26153.1	2
Paris	452	494	101926	12138.9	1
New York	379	402	71631	11594.2	1
London	340	352	57630	5122.6	1
Tehran	394	423	77421	13353.8	2

The presence of two connected components in Istanbul and Tehran introduces additional challenges, as UAV coverage strategies must account for disjoint sub-networks when planning efficient routes.

C. Simulation Results and Analysis

The simulations analyzed UAV performance using predefined parameters, The mission parameters are set as: $S_{svc} = 5$ m/s, $S_{dh} = 8$ m/s, $T_{lim} = 1800$ s, $P_s = 130$ W, $P_{dh} = 195$ W , reflecting realistic UAV operations similar to commercial drones such as the DJI Phantom 4 [2] [3]. Energy consumption was computed as the product of power and time, converted into watt-hours (Wh).

To ensure comprehensive analysis, the simulation considers required and non-required edges, assigning service costs for required edges and deadhead costs for non-required travel. The energy consumed was calculated as the product of power and time, converted to watt-hours (Wh) for each operational mode. The algorithms were evaluated on diverse city datasets, each containing unique edge distributions, to test the adaptability and scalability of the approaches. The Dijkstra algorithm was used for pathfinding, enabling UAVs to follow the shortest routes to reduce unnecessary travel.

These simulation settings provide a consistent framework for comparing the Greedy and multiple-sensors algorithms, allowing for meaningful analysis of energy consumption, UAV count, and overall efficiency.

The results are summarized in Table II and Figures 2 and 3. These highlight that the multiple-sensors algorithm consistently reduced the number of UAVs required while maintaining efficient coverage, albeit with slightly higher per-UAV energy usage due to extended mission times. The Greedy algorithm, while simpler, required more UAVs to achieve complete coverage, but distributed the workload more evenly among them.

Table II summarizes the costs associated with UAVs and sensors for two algorithms: Greedy and UAV with Multiple Sensors. Based on recent data from the Amazon website, each UAV is assumed to be a DJI Phantom 4, priced at \$1300. In the Greedy algorithm, a total of 5 UAVs are required, leading to a UAV cost of \$6500. The associated sensor cost amounts to \$65,000, resulting in the Greedy algorithm achieving the lowest total cost of \$71,500.

In contrast, the UAV with Multiple Sensors algorithm minimizes the number of UAVs required to just 2 units, reducing the UAV cost to \$2600. However, this advantage is offset by a substantial increase in sensor costs to \$130,000, as each UAV must simultaneously carry all sensor types. This configuration drives the total cost up to \$132,600, making it the most expensive option despite the smaller fleet size.

It is worth noting that while the DJI Phantom 4 may not be practical for carrying multiple sensors simultaneously, the simulation results suggest that even if replaced by a heavier-duty UAV such as the Freefly Alta X, priced at approximately \$15,000, the UAV with Multiple Sensors algorithm would still remain the most expensive approach due to the combined UAV and sensor costs. This highlights the trade-off between minimizing UAV fleet size and incurring higher sensor integration costs.

Figure 2 illustrates the total energy consumption of UAVs for each algorithm as the number of sensors per UAV increases, specifically for the New York City dataset. The Greedy algorithm exhibits noticeably higher energy consumption compared to the UAV with Multiple Sensors algorithm. This elevated energy usage in the Greedy approach likely results from its less optimized route selection, which can lead to longer travel paths and inefficient coverage. Although it performs better than the most inefficient strategies, the Greedy algorithm still requires substantial energy across all sensor configurations, indicating limitations in its adaptability as the sensor load increases.

In contrast, the UAV with Multiple Sensors algorithm consistently demonstrates superior energy efficiency regardless of the number of sensors installed. Its ability to integrate multiple sensing tasks within a single flight path minimizes redundant travel and reduces overall energy demand. This efficiency becomes especially clear as the number of sensors increases, where other algorithms would typically experience escalating energy costs. These results suggest that the UAV with Multiple Sensors algorithm not only leverages sensor availability more effectively but also offers a scalable solution for energy-constrained UAV operations in complex urban environments such as New York City.

Figure 3 presents the total energy consumption of two UAV algorithms—Greedy and UAV with Multiple Sensors—across five urban datasets: Istanbul, London, New York, Paris, and Tehran. The Greedy algorithm demonstrates moderate performance but still results in substantial energy usage, likely due to less optimized route selection. In contrast, the UAV with Multiple Sensors algorithm consistently achieves the lowest energy consumption across all cities and remains efficient even as the number of sensors per UAV increases. This highlights its ability to leverage multiple sensors on a single UAV to minimize travel costs while fulfilling coverage requirements, underscoring its advantage for energy-constrained UAV operations across diverse urban environments. It should be noted that this evaluation assumes fixed sensor configurations and static mission settings. An important extension for future work is to incorporate dynamic mission scenarios and adaptive payload strategies, where UAVs can reconfigure their sensing capabilities in real time to better match changing requirements.

TABLE II: Costs of Sensors and UAVs

Algorithm	UAVs Used	Sensor Usage	Sensor Cost ($)	UAV Cost ($)	Total Cost ($)	Lowest Cost
Greedy	5	lidar (1), ultrasonic (1), rgb camera (1), infrared (1), radar (1)	65,000	6,500	71,500	True
Multiple Sensors	2	rgb camera (2), ultrasonic (2), radar (2), lidar (2), infrared (2)	130,000	2,600	132,600	False

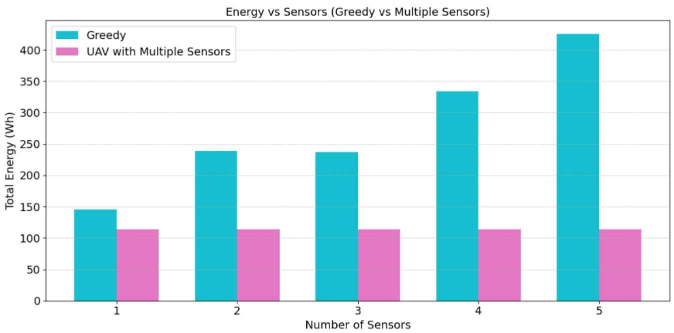

Fig. 2: New York City

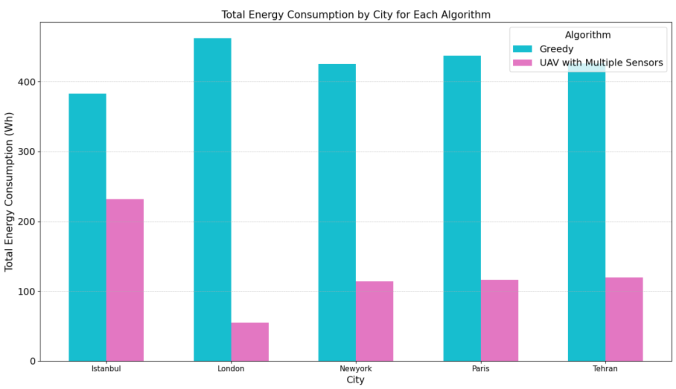

Fig. 3: Five Different Cities

VI. Conclusion

The findings highlight the critical trade-offs between UAV count, energy consumption, and operational cost in solving the line coverage problem. The Greedy algorithm demonstrates an effective balance between cost and efficiency, making it suitable for scenarios where both performance and resource usage are equally important. By assigning each UAV a single sensor and dynamically selecting compatible edges, it minimizes equipment cost while maintaining operational flexibility. In contrast, the Multiple Sensors approach reduces the overall fleet size by equipping each UAV with all required sensors, enabling them to perform diverse tasks without assignment mismatches. This significantly improves coverage efficiency and reduces idle time, but comes at the cost of higher energy demands and increased sensor-related expenses.

Together, these strategies establish a scalable framework for UAV deployment that can be adapted to diverse environments and mission priorities. They offer practical guidance for real-world applications such as urban planning, disaster response, and smart city operations. By optimizing UAV allocation and utilization, this work contributes to the development of more efficient, flexible, and sustainable aerial coverage solutions. Future research will extend this framework to dynamic mission scenarios and adaptive payload strategies, where UAVs reconfigure sensors in real time. These directions would improve robustness and expand applicability to disaster response and other time-varying environments.

Acknowledgment

This research was partly supported by the Global Scholarship Program for Foreign Graduate Students at Kookmin University in Korea, by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2022R1F1A1074672), and by a grant (22183MFDS436) from Ministry of Food and Drug Safety.

References

[1] A. Otto, N. A. H. Agatz, J. F. Campbell, B. Golden, and E. Pesch, "Optimization approaches for civil applications of unmanned aerial vehicles (uavs) or aerial drones: A survey," *Networks*, vol. 72, pp. 411–458, 2018.

[2] S. Agarwal and S. Akella, "Line coverage with multiple robots," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 3248–3254.

[3] ——, "Line coverage with multiple robots: Algorithms and experiments," *IEEE Transactions on Robotics*, vol. 40, pp. 1664–1683, 2024.

[4] E. Balestrieri, P. Daponte, L. De Vito, F. Picariello, and I. Tudosa, "Sensors and measurements for uav safety: An overview," *Sensors*, vol. 21, no. 24, p. 8253, 2021.

[5] M. Mitschke, N. Uchiyama, and O. Sawodny, "Online coverage path planning for a mobile robot considering energy consumption," in *2018 IEEE 14th International Conference on Automation Science and Engineering (CASE)*, 2018, pp. 1473–1478.

[6] M. Luo, X. Hou, and J. Yang, "Surface optimal path planning using an extended dijkstra algorithm," *IEEE Access*, vol. 8, pp. 147 827–147 837, 2020.

[7] J. Zhang and H. Huang, "Occlusion-aware uav path planning for reconnaissance and surveillance," *Drones*, vol. 5, no. 3, p. 98, 2021.

[8] B. Dai, Y. He, L. Yang, Y. Su, Y. Yue, and W. Xu, "Simsf: A scale insensitive multi-sensor fusion framework for unmanned aerial vehicles based on graph optimization," *IEEE Access*, vol. 8, pp. 118 273–118 284, 2020.

[9] J. Li, Y. Xiong, J. She, and M. Wu, "A path planning method for sweep coverage with multiple uavs," *IEEE Internet of Things Journal*, vol. 7, pp. 8967–8978, 2020.

[10] J. Munoz, B. Lopez, F. Quevedo, C. A. Monje, S. Garrido, and L. E. Moreno, "Multi-uav coverage path planning in urban environments," *Sensors*, vol. 21, no. 21, p. 7365, 2021.

[11] S. Agarwal and S. Akella, "The single robot line coverage problem: Theory, algorithms, and experiments," *Networks*, vol. 82, pp. 479–505, 2022.

[12] J. F. Campbell, A. Corberan, I. Plana, and J. Sanchis, "Drone arc routing problems," *Networks*, vol. 72, no. 4, pp. 543–559, 2018.