On WVSN Deployment and Scheduling for 3D Indoor Monitoring: A Cross-Layer Design Approach

Zhonghui Wang
Louisiana State University Shreveport
Shreveport, LA, USA
Email: zhonghui.wang@lsus.edu

Feng Wang
University of Mississippi
University, MS, USA
Email: fwang@olemiss.edu

Khaled Sabbahein
Mississippi Valley State University
Itta Bena, MS, USA
Email: Khaled.sabahein@mvsu.edu

***Abstract*—Wireless Visual Sensor Networks (WVSNs) are widely used but require effective strategies for coverage, connectivity, and efficient data collection due to high visual data demands. This paper adopts a cross-layer approach, framing sensor deployment and scheduling as an optimization problem. We propose SEED, a Scheduling-aware Enhanced Depth-first search algorithm, capable of achieving optimal solutions given sufficient time. Evaluations show our approach reduces communication latency and minimizes sensor usage effectively.**

I. INTRODUCTION

Visual sensor networks (VSNs) are integral in domains like environmental surveillance, military operations, healthcare, and security [1], [2]. Deployed in 3D indoor spaces, these sensors, often mounted on walls or ceilings, track actions within designated areas. Examples include retail applications like Amazon Go and Alibaba Tao Cafe [1], where visual sensors streamline tasks such as inventory management, account validation, and payment processing using deep learning and data fusion. However, most current implementations rely on wired connections, incurring significant deployment and maintenance costs. Wireless visual sensor networks (WVSNs) offer a cost-effective alternative, but their unique fan- or pyramid-shaped sensing ranges [3] challenge traditional WSN deployment strategies for 3D spaces.

To optimize wireless communication, a Link layer scheduler within the MAC protocol layer ensures equitable resource utilization. However, WVSNs handle much larger visual data compared to the scalar data in WSNs, rendering existing WSN communication scheduling methods inadequate. Effective WVSN deployment must address 3D coverage, connectivity, efficient data collection, and communication scheduling, reducing overhead, latency, and extending network lifespan.

This paper employs a cross-layer design to resolve these challenges, combining sensor deployment, data streaming, and wireless link scheduling into an optimization problem. A divide-and-conquer approach addresses link scheduling via a TDMA MAC protocol using a busy link-first edge coloring algorithm, managing varied link loads effectively. Building on this, the SEED algorithm (SchEduling-awareness Enhanced Depth-first search) trims sub-optimal paths and achieves optimal solutions for cross-layer optimization.

[1]https://syncedreview.com/2018/01/22/amazon-go-vs-alibaba-tao-cafe-staffless-shop-showdown/

II. RELATED WORK

Over the past decade, WSNs have advanced various research domains with dependable and scalable technologies. Recent interest has grown in WVSNs due to their unique attributes. Despite extensive studies addressing WSN deployment [4], [5], the differences in wireless visual sensors—such as sensing range, costs, and data patterns—render existing WSN solutions inadequate for WVSNs [6].

A Depth First Search algorithm was proposed in [7] to address sensing coverage and connectivity. Chow et al. [8] developed a model for optimal angular coverage in WVSNs using a minimal sensor set. Additionally, the FoVIC heuristic maximizes coverage by resolving uncovered nodes via a mathematical model. However, these methods focus on 2D scenarios, which are less effective for 3D indoor monitoring.

To improve energy efficiency, [9] introduced a TDMA scheduling scheme that conserves power and reduces transmission delays, with minimal trade-offs in WakeUp intervals. However, these solutions, designed for WSNs and scalar data, cannot effectively address challenges associated with larger visual streaming data in WVSNs.

III. PROBLEM STATEMENT

In our 3D indoor environment, visual sensors must fully cover specific regions while being fixed to ceilings or upper walls with static orientations. The goal is to minimize the number of sensors while ensuring full coverage. Additional challenges include areas without WiFi coverage, the need to avoid WiFi interference, and maintaining connectivity to the base station through independent wireless infrastructure. Deployment must also account for emergency scenarios, like power outages, requiring strategies to extend operational lifespan when sensors rely solely on batteries.

Denote by $\mathbf{A}$ the regions requiring surveillance within the specified 3D indoor setting. Employ $\mathbf{L}$ to signify the set of potential deployment locations (e.g., ceiling and upper portions of walls) where wireless visual sensors can be positioned. A tuple (L, D) signifies the visual sensor positioned at location $L \in \mathbf{L}$ with the facing direction designated as D. To convert the continuous space in $\mathbf{A}$ and $\mathbf{L}$ into a discrete arrangement, we segment it into grids. These grids use g_A to express the granularity within $\mathbf{A}$ and g_L to convey the granularity within $\mathbf{L}$. Additionally, the surface of the facing direction sphere is divided into grids, akin to the divisions of longitudes and latitudes on Earth's surface. This grid segmentation employs g_D to denote its granularity.

We assume that time is divided into equal intervals referred to as time-frames, and each time-frame can be further split into slots of identical length referred to as time-slots. We use $slot$ to denote one single time slot. The considered wireless visual sensor network can be deemed as a graph $G = (V, E)$. Each $v \in V$ denotes a wireless visual sensor, and every $e \in E$ represents the communication link between two sensors or between a sensor to the base station. R_{max} denotes the maximum communication range achievable for each wireless visual sensor node. Our goal is thus to find the deployment plan $P = \{(L_1, D_1), (L_2, D_2), ...\}$, and the contention-free link layer scheduler. As visual data streaming is often collected in a round by round manner in WVSNs, we let the data generated at node v_i be r_i packets per round, for $i = 1, 2, ..., |V|$, and v_0 represents the base station. Let t_0 denotes the time that a data collection round starts. Define a link schedule as $Schedule = \{(l_{x_1,y_1}, t_1)\ , (l_{x_2,y_2}, t_2), ..., (l_{x_k,y_k}, t_k)\}, t_0 \leq t_1 \leq ... \leq t_k$, where (l_{x_i,y_i}, t_i) means that the link at edge (x_i, y_i) will be activated at time t_i to send a packet from sensor x_i to y_i. The deployment and traffic scheduling together can thus be formulated as a cross-layer optimization problem to find a deployment plan P and link schedule $Scheduler$, subject to the following constrains:

(1) Area Coverage Constraint:

$$\forall a \in \mathbf{A}, \exists (L, D) \in P, \text{ such that } a \in cover(L, D, R_S),$$

where R_S is the sensing range of a visual sensor and $cover(L, D, R_S)$ represents the 3D space covered by the visual sensor at (L, D).

(2) Contention-free Constraint:

$$\begin{aligned}
&\forall (l_{x_i,y_i}, t_i), (l_{x_j,y_j}, t_j) \in Scheduler, \text{ if } t_i = t_j, \\
&\text{then, } x_i \neq x_j, x_i \neq y_j, y_i \neq x_j, y_i \neq y_j, \\
&|x_i, x_j| > R_{max}, |x_i, y_j| > R_{max}, |y_i, x_j| > R_{max}, \\
&|y_i, y_j| > R_{max}.
\end{aligned}$$

where $|x, y|$ denotes the distance between sensor x and y. [2]

(3) Traffic Source Constraint:

$$r_i + \sum_{(L_{x_j,y_j}, t_j) \in Schedule} I_{[y_j = v_i]} = \sum_{(l_{x_j,y_j}, t_j) \in Schedule} I_{[x_j = v_i]};$$

(4) Traffic Destination Constraint:

$$\sum_{(l_{x_j,y_j}, t_j) \in Schedule} I_{[y_j = v_0]} = \sum_{1 \leq i \leq |V|} r_i;$$

where $I_{[\cdot]}$ is the indicator function. Constraints (3) and (4) focus on end-to-end traffic, which follow that when the data collection round finishes, each sensor must send out all its data packets and the base station must receive all of them. In order to minimize the visual data streaming delay of the entire WVSN, this sub-objective can be achieved by minimizing t_k. The rationale therein is that minimizing t_k can minimize the number of slots in a frame, which can potentially increase the throughput (i.e., allow better streaming quality) or decrease the latency to send all streaming data to the base station (i.e., minimize the streaming delay). Additionally, minimizing t_k can also help improve the reliability as in practice some saved time slots can be added back to the end of the schedule and used to rescue the occasionally lost packets. Our objectives are thus to (a) minimize $|P|$, (b) minimize the visual data streaming delay $\min\{t_k\}$.

[2]" Note that here We consider a strong contention-free requirement where a reliable data transfer protocol may be used to acknowledge the receipt of a data packet". In case of no acknowledge required, the constraint can be relaxed to only have $|x_i, y_j| > R_{max}$ and $|x_j, y_i| > R_{max}$

IV. Solution Design

We use a divide and conquer strategy to tackle the formulated cross layer optimization problem. Specifically, we present a collision-free scheduling solution named Busy Link First algorithm which can obtain the optimal result for a given traffic load. Finally, we use the scheduling solution as building block to further solve the original cross layer optimization problem.

A. Busy Link First Algorithm

For a given topology, we can apply the Edge-Coloring algorithm to group the communication links together that can carry out transmission simultaneously. Edge-coloring of a graph is an assignment of "colors" to the edges of the graph so that no two incident edges have the same color. For example, Figure 1 shows an edge coloring of a graph by the colors red, blue, and green. The edges in the graph can be deemed as wireless communication links, and the vertices can be wireless visual sensors in the WVSN. Therefore, since edge CA, EB and GD have been colored by red, the wireless communications between CA, EB and GD can be performed simultaneously, the same situation for BA and FC. However, the communication between DB has to occupy additional time interval. As such, the links with same color can transmit messages simultaneously without collision, which serves as the fundamental base of our collision-free traffic scheduling algorithm.

When we deal with scalar data captured by normal wireless sensor, it can be transferred within single time-slot for each group of communication links with same color due to the smaller size of data, even the accumulated data can be forwarded to the next hop within single time-slot via data aggregation. In such cases, the traditional TDMA solution can be directly applied, and the total number of time-slots used in each round is equal to the number of groups of communication links (i.e., the number of colors).

However, in WVSNs, we collect visual data which is much larger than scalar data in size, and some wireless visual sensors have to serve as relay node to forward visual data for other sensor nodes. Therefore, the captured visual data or accumulated data cannot be transmitted to next hop within single time-slot in each round. In order to collect all the captured visual data to the base station in each round, multiple time-slots have to be applied to some groups of communication links.

Moreover, we note that the time-slots used by each group of communication links may not be equal, where the number of time-slots for each group is decided by the busiest communication link in the group. For instance, without loss of generality, assuming in Figure 1 each visual sensor generates 1 unit of visual data in each round, and 1 unit of visual data can be transmitted to next hop within 1 time-slot. Figure 2 shows the network topology with traffic load attached beside each communication link to illustrate how much traffic may go through the corresponding link. From Figure 2, we know that communication links FC, EB, and GD transmit 1 unit of visual data in each round (i.e., 1 unit/round), DB and CA transmit 2 units/round, and BA transmits 4 units/round. According to Figure 1 which indicates the same network topology as figure 2, we know that the total communication links need to be divided into 3 groups in order to achieve collision-free transmission based on the edge-coloring solution. Since identical time interval have to be assigned to each group in traditional TDMA, 12 time-slots are needed to collect all the visual data back to the base station in each round, where 12 time-slots are calculated by 4 units/round times 3 groups, with 4 units/round having to be chosen as the number of time-slots for each group of communication links due to the traffic load of the most busy link is 4 units/round. However, according to Figure 1 and 2, since we know the maximum traffic load within each group of communication links as 2 units/round in red group, 4 units/round in green, and 2 units/round in blue group, respectively, then it is easy to calculate how much time-slots will be wasted if the traditional TDMA solution has been applied (i.e., 4 time-slots are wasted in each round as 2 time-slots in red + 2 time-slots in blue = 4 time-slots). However, if we know the exact traffic load for each communication link, the total time-slots used to collect all the visual data back to the base station in each round can be dramatically decreased, because we can customize the time-slots for each group of communication links according to the maximum traffic load in each group. This can be achieved by using our cross-layer design approach to retrieve the exact traffic load information computed by the algorithm proposed in [10] from the network layer. As a result, when we apply color assignment solution illustrated in Figure 1, 8 time-slots are needed to collect all the data in each round. The group of communication links with red color occupies 2 time-slots, the group with blue color needs 2 time-slot, and the group with green color uses 4 time-slots, respectively.

It is worth nothing that the color assignment solution for each network topology is not unique. For instance, Figure 1 and 3 are showing two color assignment solutions for the same network topology, and there may exist more color assignment solutions for this network topology. Since multiple color assignments exist for each network topology, one natural question is whether they are the same or someone is better than others in terms of shortened delay when performing data collection. According to color assignment solution presented

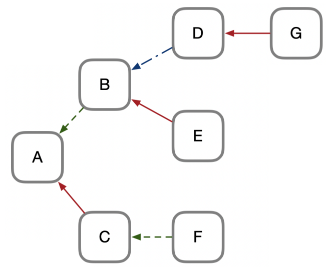

Figure 1: Example of Edge Coloring

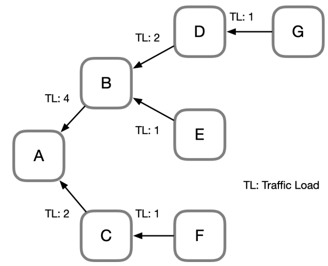

Figure 2: Network Topology with Traffic Load

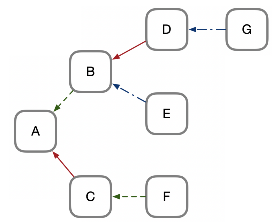

Figure 3: Example of Different Edge Coloring

in Figure 3, only 7 time-slots are needed to achieve the same goal which is even better than the solution showing in Figure 1 (8 time-slots). Therefore, different color assignment solutions may have diverse performance in terms of shortened delay. And our goal is thus to explore one of the most efficient color assignment solutions among all the combinations of communication links. To this end, we develop Busy Link First Algorithm to optimally address this problem. The Busy Link First algorithm sorts all the communication links in descending order according to their traffic load. It then picks the busiest and available communication link to assign the color first, which can generate the optimal solution in terms of minimizing the total number of time-slots as shown by the below theorem.

Theorem 1. *On any given network topology, Busy Link First algorithm can return the optimal solution to minimize the total number of required time-slots.*

Due to the space limitation, we omit the detailed proof here, which can be essentially proved via contradiction. Algorithm 1 summarizes our Busy Link First Algorithm, which takes a set of communication links with traffic load on each link. Then, according to the traffic load of each communication link, the communication link set is sorted in descending order (line 1). Within the **while** loop, the busiest and available edge is chosen and assigned a free color (line 4-10). For each set of communication links with the same color, the algorithm finds out the busiest link within each color and put the link into a set B (line 12-16). Thereafter, the number of time-slots for each set of communication links (i.e., the communication links have been assigned with same color) is assigned according to the traffic load of the busiest link within the corresponding set. Finally, the optimal time-slots assignment solution S is returned.

Algorithm 1 Busy Link First Algorithm

Input: A graph $\mathbf{G}$. A set of communication links with traffic load as edge set $\mathbf{E}(\mathbf{G})$

Output: Time-slots assignment solution S

```
Sort E(G) according to traffic load in descending order.
Let U := E(G)
while U ≠ ∅ do
    Let (u, v) be most busy edge in U.
    Let F[1 : k] be a maximal fan of u starting at F[1] = v.
    Let c be a color that is free on u and d be a color that is free
    on F[K].
    Invert the cd_u path.
    Let w ∈ V(G) be such that w ∈ F, F′ = [F[1]...w] is a fan
    and d is free on w.
    Rotate F′ and set c(u, w) = d.
    U := U − {(u, v)}
end while
for color c in all assigned colors do
    for edge e in E(G) do
        if color of e is c and traffic load is larger than S(c) then
            update S(c)
        end if
    end for
end for
return S
```

B. SchEduling-aware Enhanced Depth-first search Algorithm

With the Busy Link First algorithm as building block, we propose a visual sensor deployment solution named SchEduling-aware Enhanced Depth-first search algorithm (SEED) which can obtain optimal solution if given enough time. In contrast to conventional Depth First Search (DFS) algorithms that meticulously explore every search branch, the SEED algorithm employs branch pruning. This technique strategically eliminates candidate location search branches that are confirmed to be incapable of enhancing the existing optimal solution. As a result, the algorithm effectively reduces the scope of explored solution space. Algorithm 2 encapsulates the SEED algorithm, with the variable $depth$ signifying the count of employed visual sensors for covering the space within the ongoing search branch, while $best$ represents the currently identified optimal solution. Our initial improvement involves evaluating the feasibility of the current search branch against the $best$ solution already obtained. (line 2). Subsequently, a determination is made regarding whether to update the $best$ solution or truncate the ongoing branch. In the event that the present solution surpasses the $best$ solution, the set of final solutions $S_{min_{result}}$ is cleared, and the Busy Link First algorithm is implemented to compute the minimum count of time slots based on the prevailing network topology.

Meanwhile, the currently $best$ found solution is added into $S_{min_{result}}$ (line 6-11). For an additional enhancement in SEED's efficiency, our second refinement involves selecting a location during each iteration. This is achieved by arranging all remaining $L \in \mathbf{L}$ in descending order, relying on the maximum count of unexplored vertices (i.e., points not covered by previously placed visual sensors) that can be encompassed by a visual sensor positioned at the respective location. The approach for handling the produced facing direction follows a comparable methodology. (line 13-21). This additional step additionally diminishes the probability of opting for unviable solution choices, expediting the identification of more favorable potential locations. Furthermore, we guarantee within the algorithm that every candidate location featured in the list maintains a communication path with the base station (line 24). The subsequent enhancement is governed by the subsequent constraint: $\frac{F}{F_c^1} + \text{depth} < \text{best}$, where F signifies the remaining unexplored vertices ready for coverage, and F_c^1 indicates the count of vertices covered by the first visual sensor in the ongoing solution. The choice of F_c^1 as the denominator is grounded in the fact that, since all candidate locations are sorted before selection in descending order, the number of unexplored vertices covered by the first visual sensor is maximized compared to others. This safeguard ensures that the optimal solution is not inadvertently pruned. The purpose of this constraint is to confine the presently estimated branch. It establishes a threshold whereby the lower bound of the count of candidate locations required to encompass the remaining unexplored vertices, along with the current $depth$, must be less than the already discovered $best$ solution. If this condition is not met, the branch is pruned.(line 25-30).

V. Performance Evaluation

We evaluated our solutions using a custom JavaScript simulator to emulate 3D deployment of wireless visual sensor nodes. A baseline algorithm was developed for comparison, using a random approach to select sensor locations and optimize orientation greedily. This method, repeated 100 times, excludes connectivity considerations to minimize sensor use. Results show SEED outperforms the baseline by 64%.

Figure 3 demonstrates that larger 3D spaces require more time slots for data transmission. Our customized TDMA without sorting reduces time slots by 25% compared to the original TDMA, while the Busy Link First algorithm improves this further by 29%.

With a 1-hour constraint, SEED achieved optimal results for 3D spaces of lengths 100 and 200, suggesting better outcomes with enhanced computational resources and time. SEED also identifies multiple deployment sets, with Figure 5 showing up to 46% disparity between best and worst cases in time slot usage. These findings highlight the value of our cross-layer approach for data delivery and link scheduling optimization.

VI. Conclusion

This paper addresses the scheduling-aware deployment of WVSNs for 3D indoor monitoring using a cross-layer design. A customized TDMA MAC protocol, Busy Link First—an extension of the Directed Edge Coloring algorithm—ensures collision-free scheduling. Simulations show that Busy Link First reduces required time slots by 39% compared to the original TDMA solution (Figure 4). Building on this, we propose SEED (SchEduling-aware Enhanced Depth-first search) to optimize WVSN deployment in 3D spaces, delivering optimal solutions with sufficient time.

Algorithm 2 SchEduling-aware Enhanced Depth-first search Algorithm

Input: **L**, **A**, D
Output: Set of minimized deployment solutions $S_{min_{result}}$
$Initialize$: List of **A**, D, **L** and $best$
DFS($depth$)
if $depth \geq best$ **then**
  return
end if
if A == 0 **then**
  if $depth < best$ **then**
    clear $S_{min_{result}}$
  end if
  update $best$
  $min_{result}.timeSlots \leftarrow BusyLinkFirst$
  add $min_{result} \rightarrow S_{min_{result}}$
else
  for all C_L in **L do**
    compute all $face \in D$ for C_L
    select $face$ with max F_c
  end for
  sort $tempList$ by descending order according F_c
  for all $rank$ (i.e., all C_L with the same max F_c) in $tempList$ **do**
    resort each $rank$ by ascending order
  end for
  store $tempList \rightarrow Queue$
  while $Queue \neq \emptyset$ **do**
    $C_L \leftarrow Dequeue$
    if C_L within R_{max} of any deployed visual sensor **then**
      if $lowerBound(F)$ + $depth \geq best$ **then**
        $break$
      end if
      record C_L* and F_c* then remove them
      DFS($depth$+1)
      add F_c* back to **A**
    end if
  end while
  add all removed C_L* back
end if
return $S_{min_{result}}$

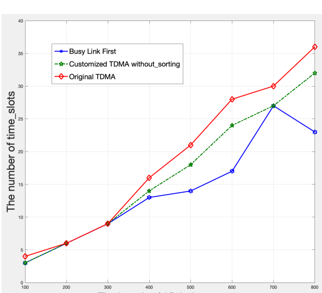

Figure 4: Comparison among three TDMA solutions

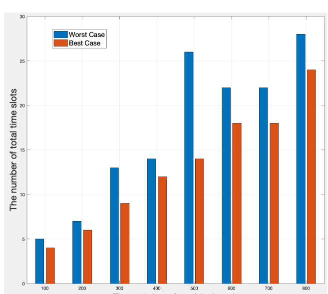

Figure 5: Worst and best case total time slots needed among deployments minimizing required visual sensors for different corridor length

VII. Acknowledgments

This research is partially supported by an NSF I/UCRC Grant (1822104).

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer networks*, vol. 38, no. 4, pp. 393–422, 2002.
[2] C.-F. Huang and Y.-C. Tseng, "A survey of solutions to the coverage problems in wireless sensor networks." *Journal of Internet Technology*, vol. 6, no. 1, pp. 1–8, 2005.
[3] S. Xu, W. Lyu, and H. Li, "Optimizing coverage of 3d wireless multimedia sensor networks by means of deploying redundant sensors," *arXiv preprint arXiv:1509.04823*, 2015.
[4] A. Raha, S. Maity, M. K. Naskar, O. Alfandi, and D. Hogrefe, "An optimal sensor deployment scheme to ensure multi level coverage and connectivity in wireless sensor networks," in *2012 8th International Wireless Communications and Mobile Computing Conference (IWCMC)*. IEEE, 2012, pp. 299–304.
[5] X.-Y. Li, P.-J. Wan, and O. Frieder, "Coverage in wireless ad hoc sensor networks," *Computers, IEEE Transactions on*, vol. 52, no. 6, pp. 753–763, 2003.
[6] S. Megerian, F. Koushanfar, M. Potkonjak, and M. B. Srivastava, "Worst and best-case coverage in sensor networks," *Mobile Computing, IEEE Transactions on*, vol. 4, no. 1, pp. 84–92, 2005.
[7] G. Sheramin, N. Ebrahimian, A. H. Navin, and M. K. Mirnia, "Connectivity issue in wireless sensor networks by using depth-first search and genetic algorithm," in *Computational Intelligence and Communication Networks (CICN), 2010 International Conference on*. IEEE, 2010, pp. 377–381.
[8] K.-Y. Chow, K.-S. Lui, and E. Y. Lam, "Maximizing angle coverage in visual sensor networks," in *Communications, 2007. ICC'07. IEEE International Conference on*. IEEE, 2007, pp. 3516–3521.
[9] N. A. Pantazis, D. J. Vergados, D. D. Vergados, and C. Douligeris, "Energy efficiency in wireless sensor networks using sleep mode tdma scheduling," *Ad Hoc Networks*, vol. 7, no. 2, pp. 322–343, 2009.
[10] Z. Wang, F. Wang, T. Brown, J. Xue, and J. Zhang, "Traffic aware wireless visual sensor network deployment for 3d indoor monitoring," in *ICC 2019-2019 IEEE International Conference on Communications (ICC)*. IEEE, 2019, pp. 1–6.
[11] T. Brown, Z. Wang, T. Shan, F. Wang, and J. Xue, "Obstacle-aware wireless video sensor network deployment for 3d indoor monitoring," in *GLOBECOM 2017-2017 IEEE Global Communications Conference*. IEEE, 2017, pp. 1–6.
[12] D. S. Deif and Y. Gadallah, "Classification of wireless sensor networks deployment techniques," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 2, pp. 834–855, 2013.
[13] A. Newell, K. Akkaya, and E. Yildiz, "Providing multi-perspective event coverage in wireless multimedia sensor networks," in *IEEE Local Computer Network Conference*. IEEE, 2010, pp. 464–471.
[14] X. Zhu and M. Zhou, "Multiobjective optimized deployment of edge-enabled wireless visual sensor networks for target coverage," *IEEE Internet of Things Journal*, 2023.
[15] A. Aliouat, N. Kouadria, S. Harize, and M. Maimour, "Multi-threshold-based frame segmentation for content-aware video coding in wmsn," in *Advances in Computing Systems and Applications: Proceedings of the 5th Conference on Computing Systems and Applications*. Springer, 2022, pp. 337–347.