The Thirty-First International Florida
Artificial Intelligence Research Society Conference (FLAIRS-31)

A Complete Coverage Algorithm for 3D Structural Inspection Using an Autonomous Unmanned Aerial Vehicle

Venkat Garlapati, Prithviraj Dasgupta
Computer Science Department
University of Nebraska, Omaha

Abstract

This paper presents a novel algorithm for complete coverage of three-dimensional structures to address the problem of autonomous structural inspection using an Unmanned Aerial Vehicle (UAV). The proposed approach uses a technique of cellular decomposition based on Morse decomposition to decompose the 3D target structure into 2D coverable faces that are subsequently connected using a graph-based representation. We then use graph traversal techniques such as the Traveling Salesman Problem (TSP) to generate a flight coverage path through the decomposed faces for a UAV to completely cover the target structure, while reducing the coverage time and distance. Experimental results show that our approach guarantees complete coverage of the target structure.

Introduction

The problem of structural inspection requires completely covering the exposed surface of a structure such as a building, tower, or a bridge with a sensor, such as a camera or LIDAR, to record data from the surface for detecting potential structural problems including cracks, fissures or fractures. Recently, researchers and practitioners have proposed using autonomous or semi-autonomous unmanned aerial vehicles(UAVs) as a fast, efficient and reliable means for structural inspection via 3D coverage of structures using on-board sensors (Murphy et al. 2011). However, there are several challenges that need to be addressed in 3D UAV coverage such as developing an efficient 3D coverage algorithm that guides the UAV's trajectory along the surfaces of complex structures, maneuvering the UAV autonomously in small spaces close to the structures being inspected and avoiding collisions with obstacles along the structure. We propose a UAV-based 3D coverage approach that takes the bounding coordinates of a 3D target structure and decomposes it into non-overlapping, rectangular 2D surfaces or cells using a cellular decomposition technique. The decomposed 2D cells are then modeled as a weighted graph with cells representing the graph vertices and the boundary between adjacent cells as edges with the edge weight corresponding to the distance between the centroids of two adjacent cells. A graph traversal algorithm, like the traveling salesman problem (TSP) is applied on this graph to determine the shortest possible route connecting all the graph vertices such that each vertex is visited at least once and the route terminates at the starting cell. Once the route is planned, the UAV is provided the coordinates of waypoints along this route and it visits those waypoints sequentially to cover the structure. To verify our approach, we validated its performance within the Robot Operating Systems (ROS) Gazebo simulator using a simlated Ascted Firefly UAV with different 3D structures placed in various environment layouts. Our results show that the proposed approach guarantees complete coverage of the target structures. Our TSP-based coverage approach performed up to 50% better in reducing the flight path and 12% less coverage duration than a largest-area-first approach.

Related Work

Coverage path planning (CPP) determines a path for a robot to follow to ensure that it is able to cover every point in a given environment using its on-board sensors. Researchers have proposed several algorithms to solve the robot coverage problem. Choset (Choset 2000) proposed one of the earliest and most successful techniques called Boustrophedon Cellular Decomposition (BCD), to solve CPP in a two-dimensional, planar environment. In BCD, the environment is dynamically divided into polygon-shaped cells and each cell is covered by the robot using back-and-forth sweeping motions. Later Acar *et al.*, (Acar et al. 2002) generalized the BCD by proposing a cellular decomposition approach based on critical points of Morse functions. In (Englot and Hover 2013), Englot and Hover described a sampling-based coverage path planner that finds a minimum cost path to cover the surface of a structure along with quantitative bounds on the probability of obtaining a feasible path, for an autonomous ship hull inspection problem. Recently, with the availability of affordable UAVs, several UAV-based structural coverage approaches have also been proposed. Maza and Ollero (Maza and Ollero 2007) proposed a multi-UAV coverage technique where a planar target area is partitioned into smaller regions based on UAVs' relative capabilities and initial locations. A coverage scheme that approximates 2.5D urban features for the coverage surfaces using hemispherical and cylindrical primitives is proposed in (Cheng, Keller, and Kumar 2008). However, these approaches consider one

---

Copyright © 2018, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

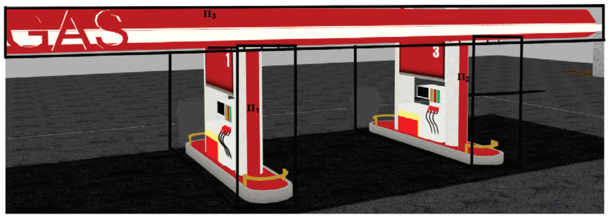


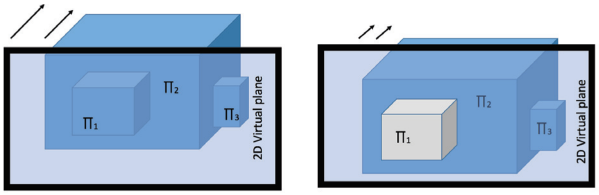
Figure 1: A sample target structure of a gas station that needs to be inspected. The solid black lines represent the envelop around the structure

dimension, that is, height or depth, as constant. Bircher *et al.* (Bircher et al. 2016) proposed an algorithm for 3D CPP for structural inspection where the structure is represented as triangular mesh, constructed from a 3D point cloud representation of the structure perceived using a depth camera. Each triangle in the mesh has a set of admissible viewpoints from which the region on the structure corresponding to the triangle can be observed by a UAV. Their propsed algorithm finds a minimal set of viewpoints that the UAV should travel through so that the structure is fully covered. In contrast to using triangular meshing on a 3D point cloud, our approach plans the coverage path along the surfaces of the structure.

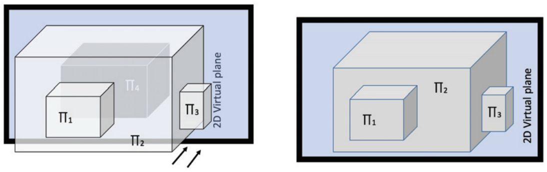
3D Cellular Decomposition

In this paper, we focus primarily on the problem of decomposing the 3D structure into 2D coverable faces. Our approach assumes that the 3D structures are enveloped by rectilinear surfaces at an offset distance from the target structure as shown in Figure 1. It is assumed that the offset distance is less than the sensing range of on-board sensors of the UAV. Every offset surface of the target structure is perpendicular to its adjacent offset surfaces. Therefore, the complete bounding structure takes the form of rectangular prisms overlapping with each other or protruding over other adjacent rectangular prisms.

A target structure is represented as a set of 3D coordinate points $\{\mathrm{x_i},\ \mathrm{y_i},\ \mathrm{z_i}\}$ representing its extremities. Each point is offset by a fixed distance d; the set of coordinate points for the enveloped target structure is given by $\{\ (\mathrm{x_i} \pm \mathrm{d}),\ (\mathrm{y_i} \pm \mathrm{d}),\ (\mathrm{z_i} \pm \mathrm{d})\ \}$. If there are holes in the target structure, a minimum clearance of $2d$ plus the width of the UAV is assumed to exist between walls on the inner surface of the hole to guarantee that the UAV can pass through them. The entire enveloped target structure is composed of one or more rectangular prisms. Each prism, π_j, is represented as $\pi_\mathrm{j} = \{\ (\mathrm{x_j}^{\mathrm{min}}, \mathrm{y_j}^{\mathrm{min}}, \mathrm{z_j}^{\mathrm{min}})\ , (\mathrm{x_j}^{\mathrm{max}}, \mathrm{y_j}^{\mathrm{max}}, \mathrm{z_j}^{\mathrm{max}})\ \}$ where $(\mathrm{x_j}^{\mathrm{min}}, \mathrm{y_j}^{\mathrm{min}}, \mathrm{z_j}^{\mathrm{min}})$ and $(\mathrm{x_j}^{\mathrm{max}}, \mathrm{y_j}^{\mathrm{max}}, \mathrm{z_j}^{\mathrm{max}})$ are the minimum and maximum coordinates of prism π_j. Hence, the enveloped target structure is represented by a collection of rectangular prisms given by $\Pi = \{\pi_1,\ \pi_2,\ \ldots\}$. The coverage surfaces of the target structure are formed by the exposed portions of these faces and the UAV's path is planned on the offset virtual surfaces on the enveloped target structure corresponding to each of the original structure's surfaces.



(a) Event - Split (b) Event - Expand



(c) Event - Contract (d) Event - End

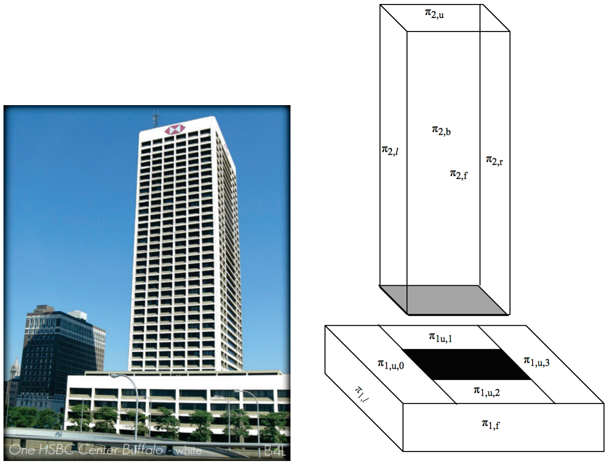
Figure 2: A Virtual Plane sweeping through the target structure showing the four events: split, expand, contract and end.

The goal of our 3D decomposition algorithm is to decompose the enveloped target structure into cells, also called faces. For this, we extend 2D Boustrophedon cellular decomposition(BCD) to handle 3D surfaces. The input to the algorithm is the set of extreme coordinate points of the rectangular prisms bounding the enveloped target structure. The algorithm first converts each rectangular prism of the enveloped target structure into a set of 2D cells. For this, a virtual 2D vertical plane $\mathcal{P}$, which is initialized with the minimum 2D coordinates $(y^{\mathrm{min}}, z^{\mathrm{min}})$ and maximum 2D coordinates $(y^{\mathrm{max}}, z^{\mathrm{max}})$ of enveloped structure, where,

$$
\begin{aligned}
y^{\mathrm{min}} &= \min_{j=1,\ldots|\Pi|} (y_\mathrm{j}{}^{\mathrm{min}}) \\
y^{\mathrm{max}} &= \max_{j=1,\ldots|\Pi|} (y_\mathrm{j}{}^{\mathrm{max}}) \\
z^{\mathrm{min}} &= \min_{j=1,\ldots|\Pi|} (z_\mathrm{j}{}^{\mathrm{min}}) \\
z^{\mathrm{max}} &= \max_{j=1,\ldots|\Pi|} (z_\mathrm{j}{}^{\mathrm{max}})
\end{aligned}
$$

$\mathcal{P}$ is moved across the enveloped structure as shown in Figure 2. As $\mathcal{P}$ moves through the enveloped structure, the connectivity changes on $\mathcal{P}$ are categorized into four events, as described below:

- $Split$, when $\mathcal{P}$ encounters the 2D faces of one or multiple prisms, rectangular holes are formed on $\mathcal{P}$. For example, as shown in Figure 2 (a), a split event occurs on $\mathcal{P}$ when it encounters the left most surface of rectangular prism π_1
- $Expand$, when $\mathcal{P}$ meets the end of smaller prism which is protruding out of larger prism and encounters the starting face of the latter larger prism. For example, as shown in Figure 2 (b), an expand event happens when $\mathcal{P}$ encounters the leftmost face of π_2 after sweeping through π_1. Another expand event happens when, while sweeping through π_2, $\mathcal{P}$ encounters the leftmost face of π_3
- $Contract$, when $\mathcal{P}$ meets the end of larger prism and encounters the starting face of smaller prism. For example, as shown in Figure 2 (c) (π_4 is a protrusion at the rear end of the structure and occluded by π_2 in Figure 2 (b)),



(a) (b)

Figure 3: (a) A Sample Real world target structure. (b) Prism π_2 creates an overlapped portion on $\pi_{1,u}$ face that is marked in black.

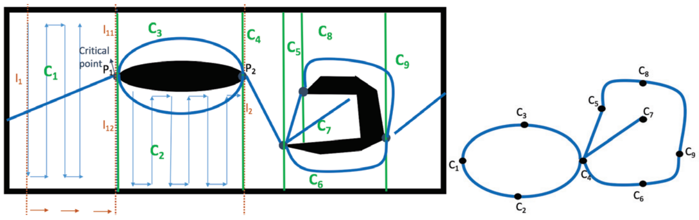
a contract event happens when $\mathcal{P}$ encounters the leftmost face of prism π_4 after sweeping through π_1, π_2, and π_3.

- End, when $\mathcal{P}$ meets the end of previous prism and does not encounter new prism. For example, as shown in Figure 2 (d), an end event happens when $\mathcal{P}$ completes sweeping all prisms π_1, π_2, and π_3 and does not encounter a new prism.

These four events are used to identify the connectivity changes on the enveloped target structure and decompose it into individual prisms. We refer to the six faces of a rectangular prism as front, left, back, right, up, and down respectively, where front is the first-encounterd face of a prism while decomposing the structure. For example, the faces of π_1, are denoted by $\pi_{1,\mathrm{f}}$, $\pi_{1,\mathrm{l}}$, $\pi_{1,\mathrm{b}}$, $\pi_{1,\mathrm{r}}$, $\pi_{1,\mathrm{u}}$, and $\pi_{1,\mathrm{d}}$, as shown in Figure 5.

Note that some of the faces of a prism might be overlapping completely or partially with a face of an adjacent prism. As an example, consider the overlap between the faces of π_1 and π_2 shown in Figure 3 (b). π_2 is protruding from π_1, and $\pi_{2,\mathrm{d}}$ is overlapped over $\pi_{1,\mathrm{u}}$. In this case, $\pi_{2,\mathrm{d}}$ is completely overlapped by a part of $\pi_{1,\mathrm{u}}$. Correspondingly $\pi_{1,\mathrm{u}}$ has a rectangular portion that is overlapped by $\pi_{2,\mathrm{d}}$, equal to the area of $\pi_{2,\mathrm{d}}$. The overlapped regions of each cell are marked in black in Figure 3 (b). The overlapped regions are inaccessible to the UAV and should not be considered for coverage. To remove the overlapped regions from the coverage path, we perform 2D Boustrophedon Cellular Decomposition (Choset 2000) on faces that has an overlapped region with adjacent faces, while considering the overlapped region as an obstacle. This decomposition results in further dividing the face into sub-faces such that all of the sub-faces are non-overlapped and accessible for coverage and the union of such cells will cover the free space of the face.

In the BCD algorithm, a virtual vertical line l is moved through the map of the environment along the x-axis. When l encounters an obstacle, e.g. at point P_1 in Figure 4 (a), its connectivity changes as it splits into two segments l_{11} and l_{12}. The location at which l's connectivity changes is called a *critical point*. Similarly, while moving l to the right when two line segments merge, e.g. at point P_2, there is, once again, a change in connectivity of l and this location is again recorded as a *critical point*. The vertical lines passing through the critical points and their endpoints on the boundary of the environment or obstacles form the boundaries of the decomposed cells. Note that the union of these cells completely cover the free space in the environment. The decomposed cells and the boundaries between adjacent cells are stored respectively as vertices and edges of a Reeb graph (Figure 4 (b)). In Figure 3 (b), after applying the 2D Boustrophedon cellular decomposition on the face $\pi_{1,\mathrm{u}}$ the decomposed sub-faces are labelled as $\pi_{1,\mathrm{u},0}$, $\pi_{1,\mathrm{u},1}$, $\pi_{1,\mathrm{u},2}$, and $\pi_{1,\mathrm{u},3}$. These sub-faces are also added into the graph to plan a tour for completely covering all the faces and sub-faces.



(a) (b)

Figure 4: 2D Boustrophedon Decomposition with cells and tour. (a) Vertical line sweeping through environment. (b)Decomposed 2D cells connected in Reeb graph.

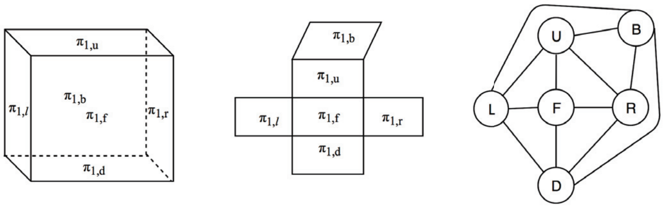


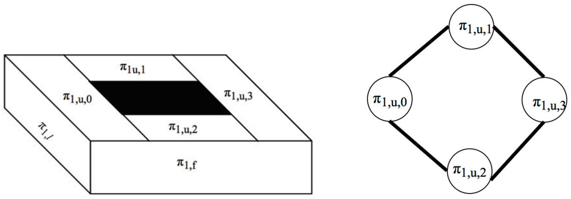
Figure 5: A Prism decomposed into 6 faces

Graph Representation of Decomposed Faces

Next, we propose an approach to connect the decomposed faces and sub-faces as a graph and determine a tour through the vertices of the graph. Each prism has six faces and each face is connected to four other faces of the same prism through a shared edge, as shown in Figure 5. We map these decomposed faces of each prism to the vertices in a weighted undirected graph $\mathcal{G}_{\mathrm{dist}}$=$\{V, E\}$, where V represents the set of decomposed faces and E represents the set of edges. Two vertices (faces) are connected by an edge if the faces are adjacent to each other and the weight of the edge is given by the distance between the centroids of the faces connected by the edge.

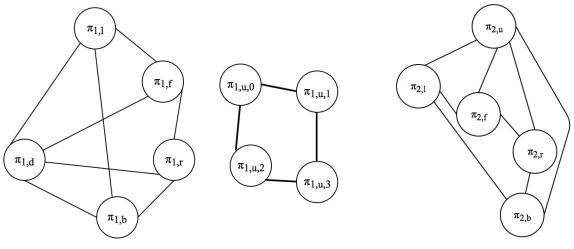
Identifying Overlapped Vertices

If the enveloped target structure corresponds to a single rectangular prism, then the corresponding coverage graph is a simple graph connecting each vertex (face) to four adjacent vertices with six vertices and twelve edges. But in case of more complex structures, where the enveloped target structure is composed of multiple prisms, we have to connect the graphs corresponding to each prism to its adjacent prisms at the appropriate adjacent faces while eliminating overlapped faces, if any. Recall, that the 2D BCD decomposition to remove overlapped regions of faces stores the decomposed sub-faces and boundaries between adjacent sub-faces as vertices and edges respectively in a Reeb graph (Figure 6 (b)). For our sample enveloped target structure shown in Figure 3 (a), $\pi_{2,\mathrm{d}}$ face is completely overlapped and it is inaccessible. Therefore, we do not include $\pi_{2,\mathrm{d}}$ in the graph representation of π_2 as shown in Figure 7 (c). In case of $\pi_{1,\mathrm{u}}$ face, we decomposed it into 2D sub-faces as it has a portion of area which is inaccessible. Hence, we remove the vertex corresponding to the face $\pi_{1,\mathrm{u}}$ as it does not exist as shown in Figure 7 (a). After removing the vertices corresponding to inaccessible faces from their respective graphs, in this case, we have a graph with three connected components where two components represent π_1 and π_2, and one more component represents the Reeb graph connecting sub-faces as shown in Figure 7.



(a) (b)

Figure 6: (a) 2D Decomposition of $\pi_{1,\mathrm{u}}$ face into sub-faces and black portion represents uncoverable areas and connecting them in graph. (b) Reeb graph connecting sub-faces of $\pi_{1,\mathrm{u}}$.



(a) (b) (c)

Figure 7: (a) Graph representation of π_1. (b) Reeb graph connecting sub-faces of $\pi_{1,\mathrm{u}}$. (c) Graph representation of π_2.

To cover sub-faces, they should be included in the coverage graph. Therefore, we need to connect the Reeb graph to the graphs representing π_1 and π_2. Each sub-face from the Reeb graph also shares its boundaries with faces of its own prism or adjacent prism or both. For example, the sub-face $\pi_{1,\mathrm{u},0}$ shares its boundaries with three faces of π_1, that are $\pi_{1,\mathrm{f}}$, $\pi_{1,l}$, and $\pi_{1,\mathrm{b}}$, one face of π_2, that is $\pi_{2,l}$, and two sub-faces $\pi_{1,\mathrm{u},1}$ and $\pi_{1,\mathrm{u},2}$ that are adjacent to it in the Reeb graph. Similarly, every sub-face shares at least two of its boundaries with two different prisms therefore, we connect π_1 and π_2 through the sub-faces. Hence, the Reeb graph representing connectivity of sub-faces, act as bridge between the graphs representing π_1 and π_2. Based on this criteria, we construct the coverage graph by connecting graphs representing π_1, sub-faces, and π_2 . After constructing all the edges to connect π_1 and π_2 through the sub-faces, the final coverage graph $\mathcal{G}_{\mathrm{dist}}$ for the target structure in Figure 3 is shown in Figure 8.

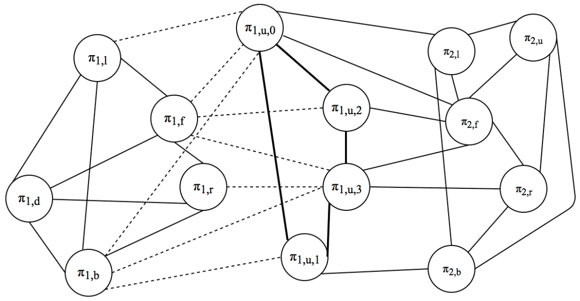


Figure 8: Final graph representing the target structure in Figure 3.

Constructing Tour

Finally, we construct a tour over the graph $\mathcal{G}_{\mathrm{dist}}$ such that the UAV is able to cover all vertices of the graph (faces of the target structure) using shortest path and covering each face at least once. This is an instance of the complete coverage path planning problem (Choset 2001) that requires constructing a tour through the graph vertices - a well known NP-hard problem (Arkin and Hassin 1994). One approach is to find the shortest tour connecting all the faces on graph $\mathcal{G}_{\mathrm{dist}}$ using TSP. This approach minimizes the face to face distance by choosing the nearest adjacent face. It also reduces the number of turns on UAV because the next nearest face to the current face is always physically adjacent on enveloped target structure. Hence, in order for the UAV to go to next face, it can take a maximum of one turn. To cover the surface of each 2D face, a lawnmower-like, back and forth pattern (Choset 2000) is used by the UAV. The width between the lanes of the back and forth motion is equal to the UAVs sensor footprint w.

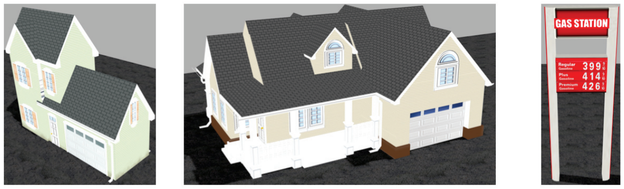
Experimental Results

To validate the suitability of our proposed 3D cellular decomposition and coverage approaches for inspection problems, we performed a series of experiments within a simulation environment with a variety of target sturctures. The simulated experiments were conducted on the RotorS Simulator (Bircher et al. 2016) using an accurate model of autonomous

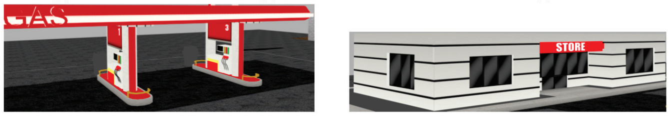
UAV called the AscTec Firefly Hexacopter. RotorS is an open source simulator that is built on Robotic Operating System (ROS) and Gazebo simulator. The simulator was run on Intel i7 8- core CPU running at 3.2 GHz machine using Ubuntu 14.04 and ROS indigo. The UAV is equipped with inertial measurement unit(IMU), generic odometry sensor, acceleration sensor, gyroscope, camera, barometric pressure sensor, and a GPS sensor. From each experiment, we have collected metrics such as number of faces, average face area of target structure, total distance covered, total time taken, and number of turns.

Each metric reveals a different aspect of the performance of our proposed approach. The number of faces and average face area are both metrics that are used to capture the complexity of the structure. More faces with low average face area implies that the enveloped target structure is more complex. The distance traveled and coverage time are both conventional metrics that are used for measuring the peformance of robotic coverage algorithms. The distance traveled during coverage is a good indication of the energy used, because the most energy intensive task for the UAV is to spin its rotors. The time taken for coverage also provides an indication of energy required to completely cover the target structure, but, unlike distance traveled, the time taken also gives us a measure of number of stops the UAV has made to take turns or change its orientation. The total number of faces is also a metric that is used to quantify the algorithm's performace. The energy consumed and time taken for coverage are proportional to the total number of turns.

For comparing the performance of our TSP-based graph traversal approach, we have used a greedy approach called Largest area first (LAF) that selects a previously uncovered face with the largest area to cover next at each step. Note that using this strategy, the face selected to cover next might not be adjacent to the face whose coverage was just completed.



(a) House1 (b) House3 (c) Name Board



(d) Gas Station (e) Store

Figure 9: Target structures used in our simulated experiments.

In Figure 10 (a), we show the total distance traversed to cover all the faces for both approaches. The greedy approach takes an average distance that is 10% more than the total distance traversed by using TSP approach. This is because, in TSP approach, the next face is selected such that it is adjacent to the current face, while in the greedy approach, the next face is not always adjacent to the current face. Figure 10 (b) shows the time taken to cover one square-meter of area verses average face area of the target structures. LAF approach requires 12% more time to cover every square meter of area when compared to the TSP approach.

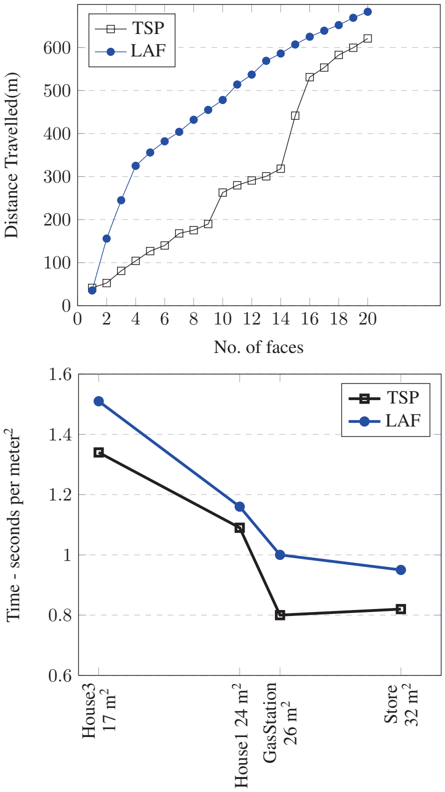


Figure 10: (a) Distance traveled as number of faces increase. (b) Time required to cover one square meter of the area.

Using TSP, the number of turns for each structure is not greater than the number of faces that the target structure has. This is because TSP selects the next face which is either on the same surface or on the surface perpendicular to the current surface, thereby reducing the number of turns. Figure 11 (a) shows that LAF approach takes an average of 30% more turns as compared to TSP. This increases UAV's energy consumption and also total time taken to cover the enveloped target structure.

Figure 11 (b) shows area coverage of each face as time increases. For example, during the inspection House1 target structure, from 77th second to 85th second, the UAV's effort is not utilized for coverage purpose. Instead, it is wasted to travel to reach the next face. Every dip in the plot to zero is an indication that the UAVs effort is wasted during that period of time. In contrast, in TSP, the next face is guaranteed to be adjacent to the current face. Hence, there is no wastage of time during the coverage.

We optimized total coverage time by choosing coverage direction perpendicular to longest edge on each face. For the Gas Station target structure, the algorithm generated 243 waypoints for the UAV as compared to only 157 waypoints with this approach, which saves 30% of the total coverage time.

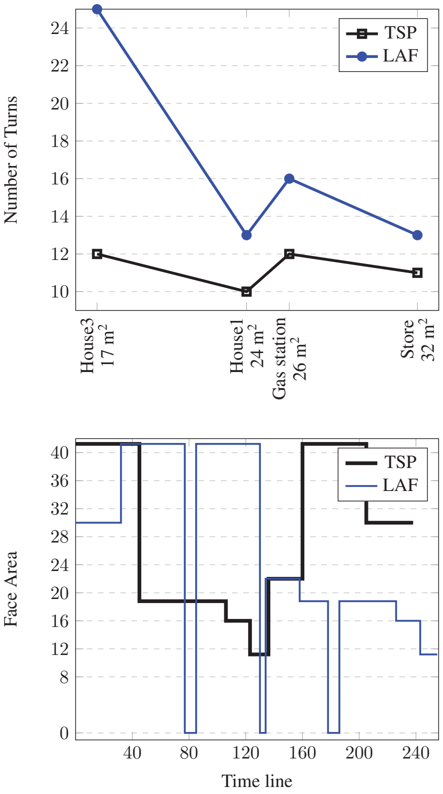


Figure 11: (a) Number of turns for each structure, (b) UAV's utilization

The target structures (House3) with small total area and more number of faces are more complex compared to the structures (Store) with large total area and less number of faces. Table 1 and Table 2 shows the results from the simulation experiments using the TSP and the largest-area-first approaches respectively. The time is measured in minutes and tour length is sum of distances between faces in final tours. Both tables show that for each target structure, TSP-based coverage perform better than largest-area-first approach.

Target Structure	Total Distance (m)	Total Time (min)	Distance b/w faces (m)	# Turns
House1	287.14	3:58	26.87	10
House3	486.11	7:45	57.43	12
Store	493.12	5:40	55.08	11
Gas Station	621.56	6:53	68.75	12
Name Board	158	1:40	12.18	5

Table 1: Different metrics of our coverage algorithm while using TSP coverage approach.

Target Structure	Total Distance (m)	Total Time (min)	Distance b/w faces (m)	# Turns
House1	294.14	4:15	35	13
House3	534.11	8:42	132.74	16
Store	557.12	6:40	121.51	12
Gas Station	683.56	8:26	143.08	25
Name Board	162	1:48	15.43	7

Table 2: Different metrics of our coverage algorithm while using Largest Area First coverage approach.

Conclusions and Future Work

The proposed 3D decomposition algorithm is thoroughly evaluated to test its capability to handle complex 3D structures. Our approach guarantees 100% coverage of a target structure and is independent of UAV platforms. This work is best suited to inspect 3D structures closely and supports UAVs with low sensor ranges. We are working towards multi-UAV coordination to inspect large structures while avoiding obstacles dynamically. In future, we would extend 3D CPP algorithm to handle curved and convex surfaces.

References

Acar, E. U.; Choset, H.; Rizzi, A. A.; Atkar, P. N.; and Hull, D. 2002. Morse decompositions for coverage tasks. 21(4):331–344.

Arkin, E. M., and Hassin, R. 1994. Approximation algorithms for the geometric covering salesman problem. 55(3):197–218.

Bircher, A.; Kamel, M.; Alexis, K.; Burri, M.; Oettershagen, P.; Omari, S.; Mantel, T.; and Siegwart, R. 2016. Three-dimensional coverage path planning via viewpoint resampling and tour optimization for aerial robots. *Auton. Robots* 40(6):1059–1078.

Cheng, P.; Keller, J.; and Kumar, V. 2008. Time-optimal UAV trajectory planning for 3d urban structure coverage. In *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2750–2757.

Choset, H. 2000. Coverage of known spaces: The boustrophedon cellular decomposition. 9(3):247–253.

Choset, H. 2001. Coverage for robotics a survey of recent results. 31(1):113–126.

Englot, B., and Hover, F. S. 2013. Sampling-based coverage path planning for inspection of complex structures. In *Proceedings of the Twenty-Second International Conference on International Conference on Automated Planning and Scheduling*, ICAPS'12, 29–37. AAAI Press.

Maza, I., and Ollero, A. 2007. Multiple UAV cooperative searching operation using polygon area decomposition and efficient coverage algorithms. In *Distributed Autonomous Robotic Systems*, volume 6. 221–230.

Murphy, R. R.; Steimle, E.; Hall, M.; Lindemuth, M.; Trejo, D.; Hurlebaus, S.; Medina-Cetina, Z.; and Slocum, D. 2011. Robot-assisted bridge inspection. *Journal of Intelligent & Robotic Systems* 64(1):77–95.