Accepted Manuscript

Model-based View Planning for Building Inspection and Surveillance Using Voxel Dilation, Medial Objects, and Random-Key Genetic Algorithm

Wei Jing, Kenji Shimada

PII: S2288-4300(17)30086-6
DOI: https://doi.org/10.1016/j.jcde.2017.11.013
Reference: JCDE 122

To appear in: *Journal of Computational Design and Engineering*

Received Date: 16 June 2017
Revised Date: 16 October 2017
Accepted Date: 29 November 2017

Please cite this article as: W. Jing, K. Shimada, Model-based View Planning for Building Inspection and Surveillance Using Voxel Dilation, Medial Objects, and Random-Key Genetic Algorithm, *Journal of Computational Design and Engineering* (2017), doi: https://doi.org/10.1016/j.jcde.2017.11.013

Model-based View Planning for Building Inspection and Surveillance Using Voxel Dilation, Medial Objects, and Random-Key Genetic Algorithm

Wei Jing[a,b,*], Kenji Shimada[a]

[a] *Department of Mechanical Engineering, Carnegie Mellon University, 5000 Forbes Ave, Pittsburgh, PA, 15213, USA*

[b] *Department of Computing Science, Institute of High Performance Computing, 1 Fusionopolis Way, 138632, Singapore*

Abstract

Model-based view planning is to find a near-optimal set of viewpoints that cover the surface of a target geometric model. It has been applied to many building inspection and surveillance applications with Unmanned Aerial Vehicle (UAV). Previous approaches proposed in the past few decades suffer from several limitations: many of them work exclusively for 2D problems, generate only a sub-optimal set of views for target surfaces in 3D environment, and/or generate a set of views that cover only part of the target surfaces in 3D environment. This paper presents a novel two-step computational method for finding near-optimal views to cover the surface of a target set of buildings using voxel dilation, Medial Objects (MO), and Random-Key Genetic Algorithm (RKGA). In the first step, the proposed method inflates the building surfaces by voxel dilation to define a sub-volume around the buildings. The MO of this sub-volume is then calculated, and candidate viewpoints are sampled using Gaussian sampling around the MO surface. In the second step, an optimization problem is formulated as (partial) Set Covering Problem and solved by searching through the candidate viewpoints using RKGA and greedy search. The performance of the proposed two-step computational method was measured with several computational cases, and the performance was compared with two previously proposed methods: the optimal-scan-zone method and the randomized sampling-based method. The results demonstrate that the proposed method outperforms the previous methods by finding a better solution with fewer viewpoints and higher coverage ratio compared to the previous methods.

Keywords: `View Planning`, `Medial Objects`, `UAV`, `Random-Key Genetic Algorithm`, `Voxel Dilation`

*Corresponding author

Email address: `wj@andrew.cmu.edu, jing_wei@ihpc.a-star.edu.sg` (Wei Jing)

1. Introduction

Building inspection and surveillance applications [1][2] with Unmanned Aerial Vehicle (UAV) have attracted increasing attentions in recent years, as a result of the rapid development of UAV technologies. The very first step of these applications is to identify an efficient plan of necessary viewpoints to achieve the required inspection and surveillance goal(s). The process of planning these viewpoints is usually termed as view planning problem, which is also referred to as sensor planning problem [3], or sensor placement problem [4] in some literatures. View planning is to find a set of viewpoints, including positions and orientations, that covers the required surface regions of the target objects [5][6] with the vision sensor carried by the robot. In the inspection and surveillance applications of view planning [1][7], the common optimization objectives are usually to reduce the number of viewpoints and to increase the coverage ratio of the surface areas. Typical vision sensors such as regular cameras, depth cameras, thermal cameras can be attached to fixed structure, mounted on a robotic arm, or mounted on an unmanned vehicle. For example, in the thermal inspection of a building [8], an UAV with a thermal camera flies around the building to check the heat loss from the building surfaces. Another similar application is the search and rescue operation in an urban area damaged by natural disaster [9], where a UAV is sent to take a set of images to look for survivors.

The view planning problem can be categorized into two groups: (1) model-based view planning, where at least a rough geometric model of the target object is available prior to the planning, and (2) non-model-based view planning, where no such model is available. In this paper, we focus on the model-based view planning problem, in particular building inspection and surveillance applications [10][11] with UAVs. As illustrated in Figure 1, building inspection and surveillance applications are the tasks which take measurement of the surfaces of buildings in a neighbourhood [12][2]. Since the geometry of the target buildings stays unchanged, the inspection and surveillance tasks are often repeated regularly, it is appropriate to formulate the building inspection and surveillance applications as a model-based view planning problem. In these applications, the number of viewpoints and the surface-coverage ratio are the major concerns, while the computational cost might be less of a concern because the planning process can be performed offline.

In this paper, we present a new two-step computational method for model-based view planning for building inspection and surveillance applications. The main contribution of our work is the usage of two geometric methods, voxel dilation and MO, as well as Gaussian sampling method to generate a set of high-quality candidate viewpoints, which leads to a set of fewer viewpoints that yields a higher coverage ratio of the surfaces of the target buildings. A combined greedy search and Random-Key Genetic Algorithm is also used to solve the formulated partial Set Covering Problem in the later step.

The rest of the paper is structured as follows. Section 2 discusses the previous work including the two methods, the optimal-scan-zone method and the sampling-based method, that are used to benchmark the proposed computational method. Section 3 defines in detail the view planning problem for building inspection and surveillance tasks, and Section 4 explains the proposed two-step computational method for finding the resultant viewpoints. Section 5 presents the results of the benchmark study along with discussions. Conclusions and recommendations for future work are presented in Section 6.

2. Previous Work

In model-based view planning, finding the near-optimal viewpoint set with a complete or high surface coverage ratio is challenging, because the search space of viewpoints is large and the coverage problem is often NP-hard. Since the 1980s researchers have proposed different approximation methods to find the solution. Most previous work used a two-step "generate-test" approach [6], with only few exceptions. The work [13] in the early 1990s is not a "generate-test" method, where the aspect visibility model is applied to find the visible regions of surface patches, and the most overlapped viewpoints are selected until full coverage is achieved. After this work, the "generate-test" approach [6] became popular. In 1995, researchers [5] introduced the measurability matrix and found the optimal solution from a candidate viewpoint set generated from the surface of a view sphere enclosing the target object. Later, researchers [14] proposed an "optimal scan zone" method that generates candidate viewpoints by directly offsetting from the sampled surface patch of the target object. The measurability matrix was also modified and the view planning problem was then formulated as a Set Covering Problem (SCP) which is solved by inter linear programming. More recently, a randomized sampling-based method [1] was proposed to generate candidate viewpoints around the target buildings, and then the greedy search algorithm was applied to solve the SCP problem. More details of the previously proposed methods and other relevant methods are described in several survey papers: [15], [6], [7], [16] and [17].

The building inspection and surveillance applications have been studied by many researchers [10][12][1]. View planning plays an important role to minimize the inspection and surveillance cost. An Unmanned Ground Vehicle (UGV) is used to monitor the buildings [10], with the viewpoint positions in 2D environment. A planning method is presented for using a UAV for surveillance tasks that only considered the top surfaces of the buildings in a 3D urban environment [18]. More recently, researchers [12] also reported an evolutionary algorithm-based view planning method also in 3D urban environment, but the method only considers covering the top surfaces of buildings, such as the roofs and the ground.

Most of the previous view planning methods for inspection and surveillance applications focus on 2D view planning or partial 3D view planning, where the poses of the viewpoints are represented with fewer than 6 Degree-of-Freedom (DOF). In addition, these methods tend to generate a greater number of viewpoints than necessary, and/or the viewpoints miss a relatively large portion of the target surfaces. In order to find a small number of viewpoints that yield a high coverage ratio, the popular two-step "generate-test" approach [6][14] can be applied, but the approach was originally designed for a single object with a relatively simple geometry. When applied to a complex target consisting of multiple buildings, the previous "generate-test" approach may not be able to generate suitable candidate viewpoints in the "generate" step. This is because the previous "generate-test" approaches use a view sphere or offsetting methods to generate viewpoints. Randomized sampling-based methods [19][20] usually work better, but they are still not effective enough in terms of space exploration.

3. Problem Formulation

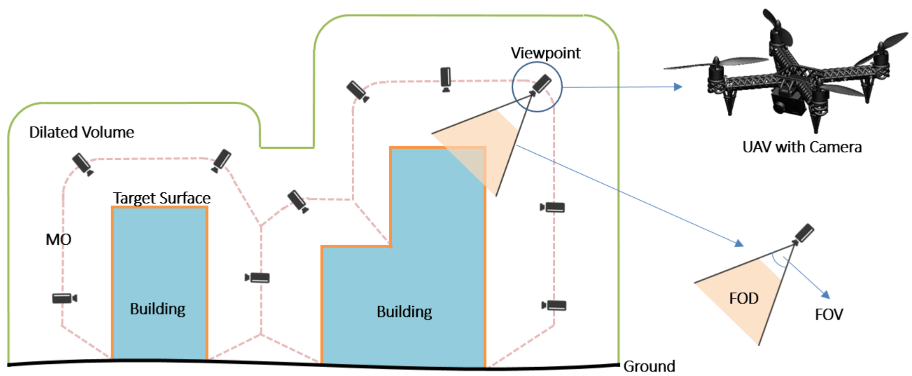


Figure 1: Building inspection and surveillance: a UAV with a vision sensor flies around buildings (shown in blue) to acquire surface information of the buildings. The vision sensor has a visible region that is defined by the field of view (FOV) and field of depth (FOD). The dilated volume is bounded by the offset surfaces of buildings (shown in green), ground (shown in black) and the buildings surfaces (shown in orange). The red dash lines illustrate the MO of the dilated volume

As illustrated in Figure 1, for the building inspection and surveillance applications discussed in this paper, a UAV flies around buildings to acquire surface information of the buildings with a vision sensor. The goal is to find a minimum set of viewpoints for the sensor so that they completely cover the surfaces of target buildings. In this application, the 6-DOF positions and orientations of a visual sensor can be achieved by the motion of the UAV and with the pan-tilt mechanism attached to the sensor. The visible region of the sensor is specified by to parameters, field of view (FOV) and field of depth. The target is a set of buildings in a city block. Since this is a model-based view planning problem, we assume that the shapes of the buildings are known and that they are represented as polygonal surface patches.

We formulate our view planning problem as a constrained combinatorial optimization problem:

$$\min \sum_{i=1}^{n} x_i \quad x_i \in \{0, 1\},$$
$$s.t. \quad S = \cup(f(p_i) \cdot x_i) \quad \text{for all i},$$

where $S = (s_1, s_1, ..., s_m)$ is the surface patches of target building; n is the number of candidate viewpoints; x_i indicates whether the candidate viewpoints should be included in the selected set of viewpoints to cover the target buildings; $x_i = 1$ means this viewpoint is included and $x_i = 0$ means it is not included. p_i is the position and orientation of the i^{th} viewpoint; function f maps a viewpoint to a set of surface patches, or a subset of S. $f(p_i)$ is the mapping that defines which surface patches are visible from viewpoint p_i, according to predefined visibility criteria. For example, if three surface patches s_5, s_6, and s_7 are visible from the third candidate viewpoint, then $f(x_3) = \{s_5, s_6, s_7\}$. In this formulation, the objective is to find a set of fewest viewpoints that covers all the surface patches of the target buildings.

4. Proposed Computational Method

In this paper, we propose a novel computational model-based view planning method for building inspection and surveillance applications. The proposed approach combines geometric methods and combinatorial optimization methods to generate view planning results with higher coverage ratio and fewer viewpoints for the coverage requirement. The first step of the proposed method uses two geometric methods: voxel dilation and Medial Object (MO), together with Gaussian sampling method to generate the candidate viewpoints. The second step finds the near-optimal subset of viewpoints from the set of candidate viewpoints by formulating the partial Set Covering Problem (SCP) and solving it with the Random-Key Genetic Algorithm and Greedy search method. Figure 2 summarizes the process flow of the proposed view planning method.

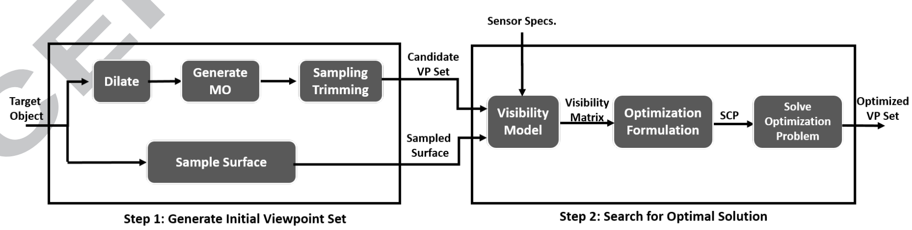


Figure 2: The proposed two-step computational method: the first step finds a set of candidate viewpoints and the second step finds the optimal subset of candidate viewpoints that covers all the surfaces of the target buildings

Compared to the previous work, [21] [14] [1], which also utilized two-step optimization consisting of the generation of candidate viewpoints and the selection of the final viewpoints. The proposed method, as described in Section 4.1, uses voxel dilation, MO and Gaussian sampling instead of simple offsetting or randomized sampling in the viewpoint generation step, yielding better results in the second step, described in Section 4.2.

4.1. Generation of Candidate Viewpoints Set

The purpose of the first step is to sample a set of candidate viewpoints with good quality. Skipping this step and trying to find an optimal set of viewpoints directly would require optimization in high dimensional space that takes impractically high computational cost. Pre-selecting candidate viewpoints narrow down the search space and reduce the computational cost significantly.

High-quality candidate viewpoints should satisfy the following conditions:

- Candidate viewpoints should have a high coverage ratio. This is because the coverage ratio of a subset of candidate viewpoints selected in the second step is capped by the coverage ratio of the candidate viewpoints.
- Candidate viewpoints should include the ones that can see reentrant edges and corners of the target buildings.
- The number of candidate viewpoints should be comparable to the number of surface patches. Too many candidate viewpoints would lead to an unnecessarily high computational cost, and too few candidate viewpoints would lead to a low coverage ratio and less optimal results.

As illustrated in Figure 3, the proposed method generates such high-quality candidate viewpoints by: (1) voxelizing the building shapes and dilating the voxelized building shapes, (2) generating MO of the dilated volume, (3) removing MO points that lie on a subset of MO defined by dilated surfaces only, and (4) sampling candidate viewpoints using Gaussian sampling on the MO and generate the viewing direction using local potential field methods. The details of these steps are explained in Section 4.1.1 to 4.1.3.

4.1.1. Binary Voxel Dilation

Binary dilation, also known as Minkowski Sum [22], is a basic operation in mathematical morphology. It is a convolution-like operation in 3D space, and generates a dilated object as illustrated in Figure 4. Dilation is widely used in computer graphics, motion planning, image processing and computational geometry. In Euclidean space, binary dilation is defined as:

$$A \oplus B = \{a + b | a \in A, b \in B\}.$$

In the proposed method, binary voxel dilation of the target object is used to generate the dilated volume (black part of right figure in Figure 4 (b)), whose MO will then be used to generate candidate viewpoint set in later steps. The dilated volume is obtained by subtracting the dilated object with the original target object.

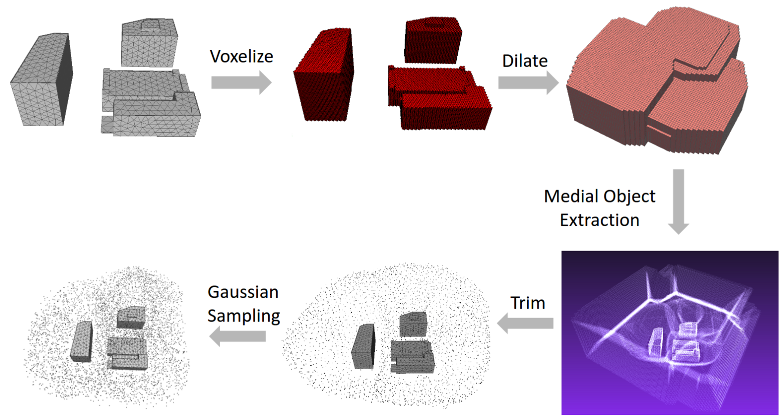


Figure 3: Generation of high-quality candidate viewpoints using voxel dilation and MO.

(a) The original model, A; and dilation mask, B.

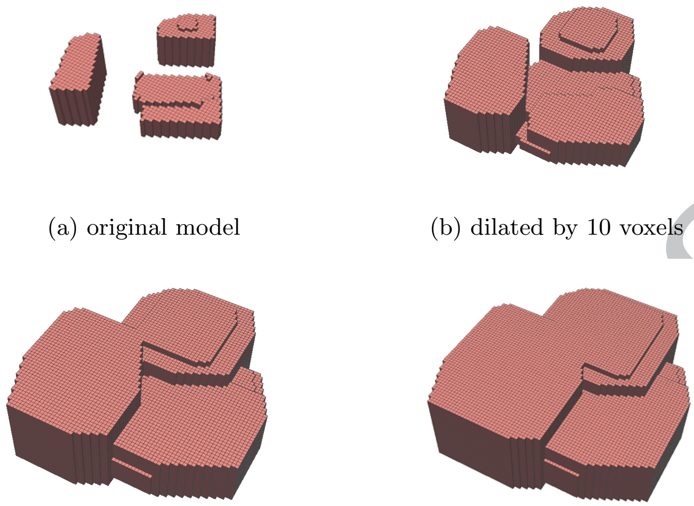
(b) The dilated volume (shown in black).

Figure 4: Example of the cross section of 3D dilation

In this paper, the building inspection and surveillance applications require to cover all model surfaces except the bottom surface. In order to incorporate better with discrete voxel space, cube is used for dilation rather than sphere in our implementation. The examples of dilated models with different dilation sizes are shown in Figure 5. The size of dilation is determined by the FOD of the vision sensor. Normally when the viewing distance is closed to the maximum of the FOD, the viewing area of sensor is maximized. In the proposed method we experimentally find that dilating our target object by about 1.4 to 1.6 times the maximum of sensor FOD yields good results. The MO generated from the dilated volume is then located at a distance of about 0.7 to 0.8 times the maximum of sensor FOD from the surfaces of the target buildings.

4.1.2. Medial Object Generation

As illustrated in Figure 6, Medial Objects (MO), also refereed as Medial Axis Transform (MAT) [23] in literature, are the skeleton of a target object, and consist of all points that have the same closest distance to at least two points on the original target object boundary [24][23]. The important property



(a) original model (b) dilated by 10 voxels

(c) dilated by 20 voxels (d) dilated by 30 voxels

Figure 5: The original and dilated objects

of MO is that it reduces the dimension of original model and preserves the geometric information of original model. The MO based methods have been used for robot pose-to-pose planning in past few years [25][26] to improve the efficient of exploring the space. However, to the best of our knowledge, it is the first time to use MO for coverage planning problems with voxel dilation and Gaussian sampling, in the UAV inspection and surveillance applications. In this paper, we use the "Skeleton Sandbox" [27] to generate MO of a voxel model. The MO is defined as:

$$\mathbb{S} = \{s \in \mathbb{V} | \exists a, b \in \delta\mathbb{V}, a \neq b, \|a - s\| = \|b - s\| = min(D(s, v(v \in \delta\mathbb{V})))\},$$

where $\mathbb{S}$ is the point set of MO; $\mathbb{V}$ is the original model; $\delta\mathbb{V}$ is the boundary of original model; $D(x, y)$ is the Euclidean distance between two points.

In order to reduce the computational cost, We uniformly resample the MO points and delete the MO points that do not associate with the surface of target object.

4.1.3. Sampling Viewpoints Around Medial Object Using Gaussian Sampling

In order to obtain the viewpoints from the MO, a sampling process is performed on the MO. We perform a sampling of Gaussian distribution with the mean value on the medial object, and a covariance matrix of $\sigma\mathbf{I} \in R^{3\times3}$. A viewpoint position is determined by randomly choosing a point on MO, and perform the Gaussian sampling strategy [28] with the mean on the chosen MO

(a) The original model (b) The MO of the original model

Figure 6: Example of Medial Object of 3D model

point. The Gaussian sampling is applied as an biased sampling methods, because the space around MO carries more meaningful geometric information of the target object, sampling biased to the MO would be more efficient compared to random sampling method. Thus better planning results are expected with Gaussian sampling method.

In addition to the viewpoint positions, the viewing direction is obtained using probabilistic local potential field method. As documented in our previous work [1], the method assume that there are attraction forces from the surface of the target buildings, and the direction of the total attraction force on a certain viewpoint is the viewing direction of that viewpoint.

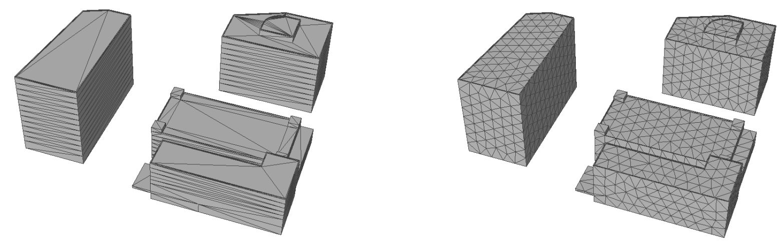
4.2. Select Viewpoints by Combinatorial Optimization

Because the visibility function that maps viewpoints to surface patches is non-parametric set-to-set mapping, the problem has to be formulated as a combinatorial optimization problem.

The previous step uses geometric information to reduce the number of viewpoints and surface patches in view planning problem to a reasonable number for the combinatorial problem in this step.

4.2.1. Subdividing the Model of Target Object

For the optimization problem formulation, another task is to subdivide the surface of the target object into a mesh of triangular surface patches, which is considered as sampling the surface. Subdividing the surface with reasonable patch size and preserve their original shape is an important task in this work; if the patch size is too large, the result may not be able to fully cover the surface because the visibility test checks the center point of the triangle only; if the patch size is too small, the computational cost can become impractically high. In this paper, we used the Bubble Mesh [29] method to generate a well-shaped triangular mesh that preserves the original shape well and maintains patch uniformity. Figure 7 shows the meshing results.



(a) The original mesh of the building (b) The uniformly subdivided mesh using Bubble Mesh method (2,404 faces and 1,258 vertices)

Figure 7: Example: Subdivision of model into a set of triangular surface patches

4.2.2. Encoding Visibility Information with Visibility Matrix

In the proposed method, a visibility model is established to decide whether a surface patch can be viewed from a viewpoint. In the model, a surface patch is visible from a viewpoint if all the following conditions are satisfied:

- The surface patch must be in the Field of View (FOV) of the sensor from the viewpoint.
- The surface patch must be in the Field of Depth (FOD) of the sensor from the viewpoint.

$$d_{min} < (\mathbf{p_{vp}} - \mathbf{p_{patch}}) \cdot \mathbf{n_{vp}} < d_{max}$$

- The viewing angle β must be within a certain range predicted by the sensor specs.

$$\beta_{min} < \arccos(\frac{(\mathbf{p_{vp}} - \mathbf{p_{patch}}) \cdot \mathbf{n_{patch}}}{\|\mathbf{p_{vp}} - \mathbf{p_{patch}}\| \, \|\mathbf{n_{patch}}\|}) < \beta_{max}$$

- There must be no occlusion, which means no solid elements lies in between the surface patch and the viewpoint. This condition is calculated using a modified version of ray-triangle intersection algorithm [30], which has $O(mn^2)$ running time, where m is the number of viewpoints and n is the number of surface patches. The algorithm connects a viewpoint and surface patch with a line segment, and checks whether the line segment intersects with other triangles of the target object.

In this paper, we use the classic pinhole camera model and projection matrix [31] to model the FOV visibility. by setting the skew parameter to 0, the ideal intrinsic matrix of a pinhole camera is:

$$\mathbf{K} = \begin{bmatrix} \alpha_x & 0 & p_x/2 \\ 0 & \alpha_y & p_y/2 \\ 0 & 0 & 1 \end{bmatrix} \in R^{3\times3},$$

where α_x, α_y are the focal length in terms of pixel dimensions on x and y axis; p_x, p_y are the number of pixels on x and y axis.

The extrinsic matrix is decided by the position $\mathbf{T}$ and orientation $\mathbf{R}$ of a given viewpoint:

$$\mathbf{E} = [\mathbf{R}, \mathbf{T}] \in R^{3\times 4}.$$

Then given the intrinsic and extrinsic matrix of camera,the projection of a point $\mathbf{p} = [x, y, z, 1]^T \in R^{4\times 1}$ on the image plane is calculated:

$$\mathbf{u} = \mathbf{K} \cdot \mathbf{E} \cdot \mathbf{p}.$$

To ensure visibility, the projection of a given point must be within the pixels range of camera. Therefore, for a given point, if $0 \leq u_1/u_3 \leq p_x$ and $0 \leq u_2/u_3 \leq p_y$ after projection, then it is within the visible region of the given viewpoint with the camera specification, otherwise it is considered as not in the visible region.

(a) (b)

Figure 8: The visibility model

The $m \times n$ visibility matrix is obtained by testing the visibility of m surface patches from n viewpoints using the above-mentioned visibility model. Each element of the visibility matrix takes 0 or 1 and indicates whether a given patch can be seen from a given viewpoint. Figure 9 shows an example of visible areas with a given viewpoint.

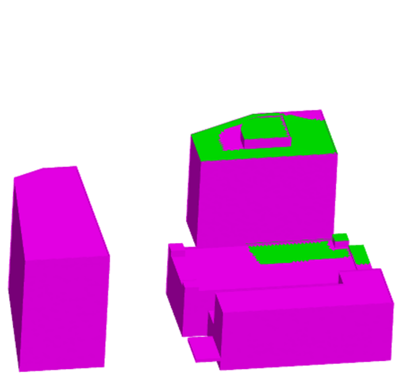


Figure 9: An example of visible areas with a given viewpoint. The visible areas are shown in green.

4.2.3. Set Covering Problem Formulation

After the visibility matrix has been generated, the SCP is formulated to find the least number of viewpoints that cover every surface patch at least once:

$$\min \sum_{i=1}^{n} x_i, \quad \text{where } x_i \in \{0,1\}, \tag{1}$$

$$s.t. \quad \mathbf{A} \cdot \mathbf{x} \geq \mathbf{c}, \tag{2}$$

where x_i is the i^{th} viewpoint, $x_i = 1$ means this viewpoint is selected and $x_i = 0$ means it is not selected; $\mathbf{A}$ is $m \times n$ visibility Matrix; the result of $\mathbf{A} \cdot \mathbf{x}$ is a $m \times 1$ vector, which indicates how many times each surface patches are covered; $\mathbf{c}$ is the measuring frequency, in this case, is $m \times 1$ vector with all 1s.

In this paper, we reformulate the SCP to a partial Set Covering Problem for practical consideration. In the partial SCP formulation, instead of full coverage, a given coverage ratio δ of the target building is required. We use $\delta = 0.99$ in the work, which indicates 99.0 % coverage of the target buildings are required. The partial SCP formulation is shown as follows:

$$\min \sum_{i=1}^{n} x_i, \quad \text{where } x_i \in \{0,1\} \tag{3}$$

$$\text{s.t.} \quad \sum_{i=1}^{m} ((\sum_{j:x_j \neq 0}^{n} A_{ij}) \geq 1) \geq \delta * m, \tag{4}$$

where constraint (4) is the partial coverage constraints, which indicates that at least $\delta \times 100\%$ of the total surface patches should be covered in the optimization formulation.

4.2.4. Solve Partial SCP Using Random-Key Genetic Algorithm and Greedy Search

Several methods can be applied to solve the Set Covering Problem and its extensions, such as Greedy search algorithm [32], Dynamic Programming [33], and Genetic Algorithm [34].

In this paper, we propose a combination of Random-Key Genetic Algorithm (RKGA)[35][36] and greedy search method for the partial SCP problem. We use real number range from 0 to 1 as random keys to encode the information. The random keys are stored in the genes of the chromosome. The decoding process is to sort the genes by value. The fitness evaluation process adds the sorted gene to the solution set one-by-one, until the coverage constraint is satisfied. An advantage of using RKGA is that the constraint is handled in the fitness evaluation, so during other GA operation such as mutation and crossover, the constraint handling is no longer necessary. The objective is to minimize the number of viewpoints required for the given coverage ratio. In this paper, a coverage ratio of 99.0% is used.

MATLAB Genetic Algorithm solver [37] has been used to solve partial SCP in this paper. The number of chromosomes is set to 2,000 and the number of iteration is set to 240. These two parameters ensures that enough computational resources are allocated to GA such that the final results would be converged and consistent. In this paper, the mutation rate is set to 0.3; and the crossover rate is set to 0.7.

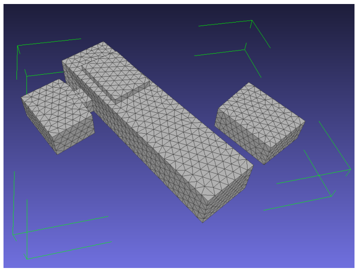
In addition to RKGA, we also make use of the Greedy search [1] results of the partial SCP. In our method, a greedy search is first performed to find a solution of the partial SCP, then initial population of RKGA is randomly sampled biased to the greedy search results. Combining the RKGA and greedy search method leads to better results and makes the RKGA converge much faster.

5. Computational Experiments, Results and Discussion

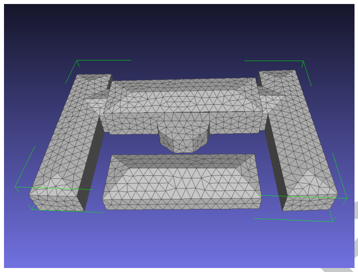
In the computational experiment, We have compared our method with two previous methods using four different sets of target buildings, with different sensor parameters. The results shown in this section demonstrate that our method outperforms the two previous methods in terms of the number of required viewpoints, as well as the coverage ratio, under different sensor parameters and different numbers of candidate viewpoints.

5.1. Setup

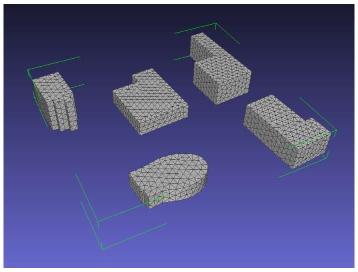
Figure 10 shows four different sets of target buildings used for validating the proposed method. We also use the sensor parameters including viewing angle and FOD. In this paper, we use 30 meters, 50 meters as the maximum FOD; and $70°, 80°$ as the maximum viewing angle in the experiments, similar to the parameters used in our previous work [1][19]. A DJI Phantom 3 camera model of $20mm$ focal length is used to model the FOV, with the following calibrated intrinsic parameters:



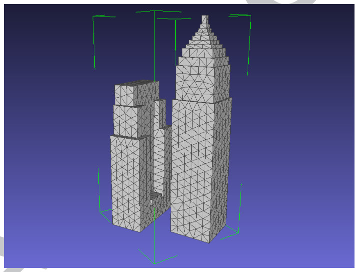
(a) Target building set 1: bounding box size is $64 \times 79 \times 13$ meters



(b) Target building set 2: bounding box size is $106 \times 71 \times 15$ meters



(c) Target building set 3: bounding box size is $85 \times 79 \times 14$ meters



(d) Target building set 4: bounding box size is $85 \times 100 \times 180$ meters

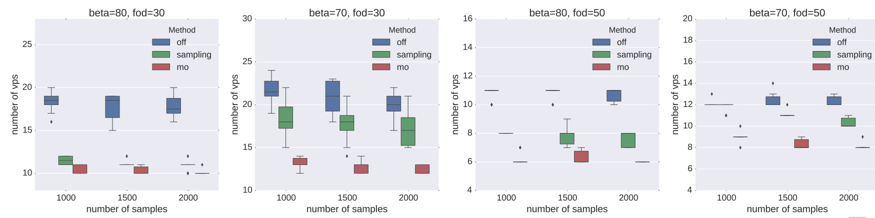
Figure 10: The polygonal geometric model and the bounding box of the target buildings

$$K = \begin{bmatrix} 2337.8 & 0 & 1991.8 \\ 0 & 2346.3 & 1466.1 \\ 0 & 0 & 1 \end{bmatrix}$$

In the computational experiments, we compare the proposed method with the two previous methods mentioned in Section 2:

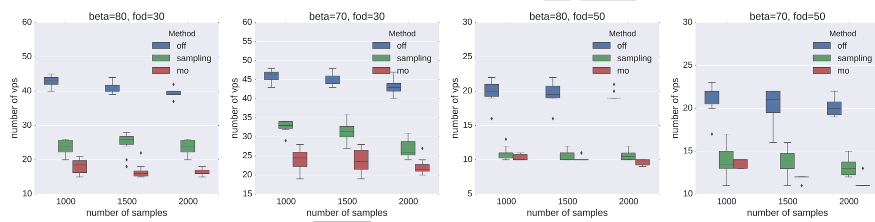
- The "optimal scan zone" or "Offset" method [14] generates a set of candidate viewpoints directly on every surface patches of the building surfaces. and offsets them with a given distance. The viewing direction is defined so that it points to corresponding surface patch.
- The randomized sampling-based method [1][19] generates a set of candidate viewpoints by randomly sampling around the target buildings, within the range of the maximum FOD.

The candidate viewpoints are generated with the three different methods mentioned above, and the same optimization method is used in the second step. During the computational experiments, we run each method with different parameters 10 times in order to show statistical comparison results.

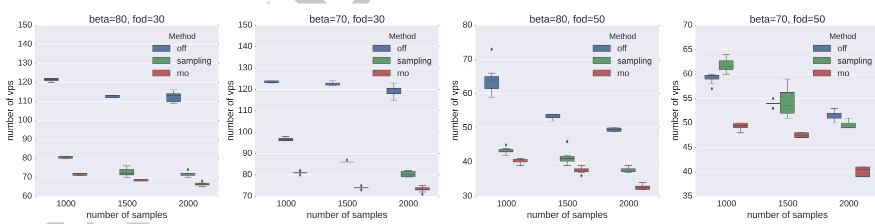


(a) Target building set 1

(b) Target building set 2



(c) Target building set 3



(d) Target building set 4

Figure 11: Number of resultant viewpoints for the required coverage. The proposed method is labelled as "mo" (shown in red); the randomized sampling-based method is labelled as "sampling" (shown in green); the "Optimal Scan Zone" or "Offset" method is labelled as "off" (shown in blue). beta is the maximum viewing angle, fod is the maximum viewing range. The y-axis is the number of required viewpoints, while the x-axis is the number of sampled candidate viewpoints

5.2. Results

Figure 11 and 12 compare the performance of our method with the two previous methods. It shows that our method consistently finds the least number of viewpoints and best coverage with different sensor parameters and different number of candidate viewpoints.

As shown in Figure 11, the proposed method finds 38.7%, 38.9%, 49.0%, 32.0% fewer viewpoints for the required coverage compared to the optimal scan zone method [14], and 19.9%, 14.4%, 17.6%, 12.1% fewer viewpoints for the required coverage compared to the randomized sampling based method [1][19] for the four target buildings respectively. Overall, the number of viewpoints required is **39.7%** less than that of optimal scan zone method and **16.0%** less than that of the recently proposed sampling-based method on average among the four sets of target buildings. The example 3D visualization of the resultant viewpoints from the proposed method and the target buildings are shown in Figure 14.

Figure 12 shows the ratios of uncovered areas of the three methods with different sensor parameters. It shows that the proposed method consistently achieves less uncovered areas, which means better coverage ratio, as compared to the other two methods. That is mainly because the proposed MO-based method explores the space more effectively with the geometric information of the target buildings.

Based on the results shown above, we conclude that the proposed view planning method has demonstrated better results with higher coverage ratios and fewer required viewpoints, compared to existing methods.

5.3. Convergence and Computational Costs

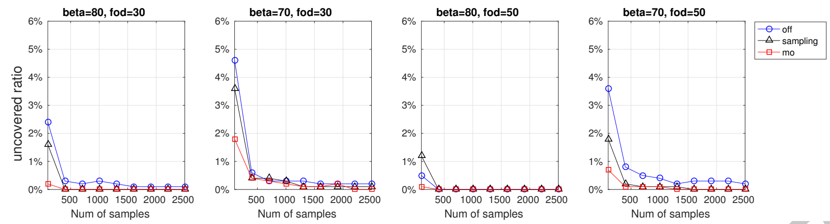
The computational time is about 20 to 30 minutes with a Core i7 computer with 8 GB RAM. The convergence rate of the RKGA is shown in Figure 13. The score of the best chromosome in a generation is shown in red, while the average score is shown in blue. It is observed the RKGA converges properly at about 150 generations.

5.4. Discussion

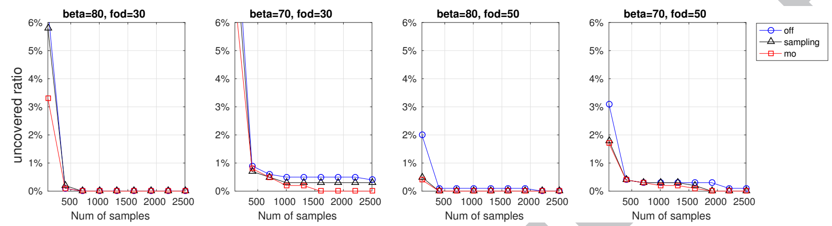
The results obtained in the previous section show that the proposed MO-based view planning method performs better than the other two methods, and there are several reasons for that.

The "optimal scan zone" or "Offset" method [14] has demonstrated good coverage ratios for most target buildings. However, the results are less optimal, because it is unable to identify and make use of important geometric features such as corners and edges, which limit its performance when there are many of those features. In addition, incomplete coverage problem still exist because invalid viewpoints may be generated by certain non-convex geometries because it is difficult to choose a good offsetting distance in those instances. The randomized sampling method usually leads to better results compared to the optimal scan zone method. However, sampling the viewpoints randomly uses less geometric information of the target buildings, which makes the candidate viewpoints generation process less effective.

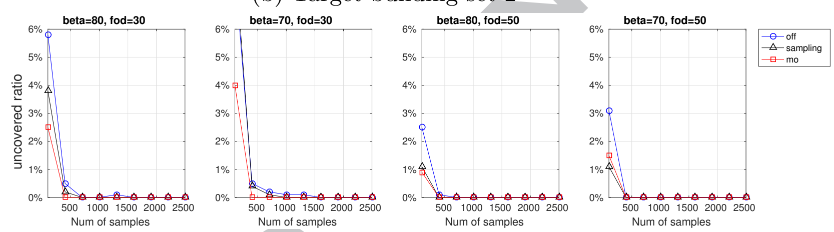
The MO-based method overcomes the above-mentioned problems. Generating candidate viewpoints by using Gaussian sampling around the MO of a dilated volume ensures that all the viewpoints obtained are valid in feasible



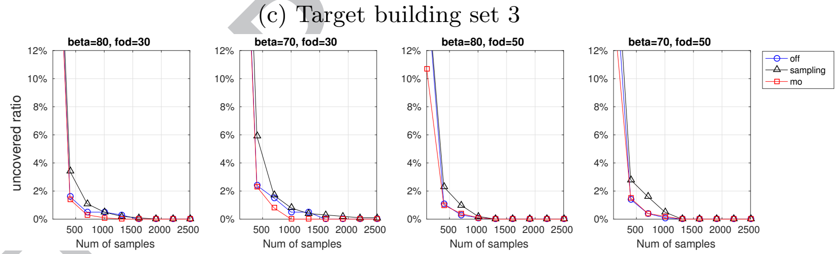
(a) Target building set 1



(b) Target building set 2



(c) Target building set 3



(d) Target building set 4

Figure 12: Percentage ratios of uncovered areas with different number of sampled candidate viewpoints. Smaller number indicates higher coverage ratio.

space. In addition, MO is the skeleton of a dilated volume, which well preserves the geometric features of target buildings, yielding better results.

6. Conclusions

This paper presents a novel model-based view planning method for UAV building inspection and surveillance applications. Voxel dilation, Medial Object (MO) and Gaussian sampling methods are used to generate the candidate

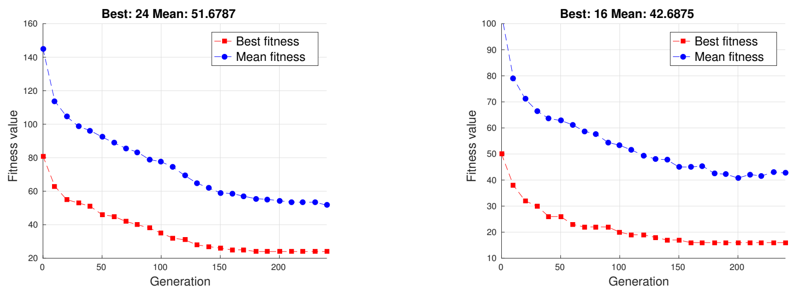
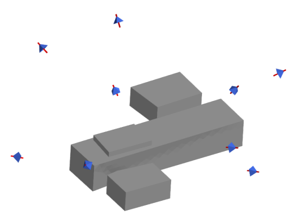
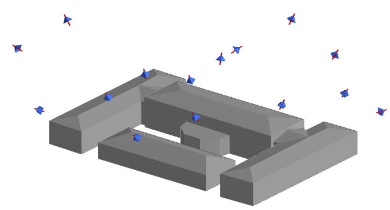


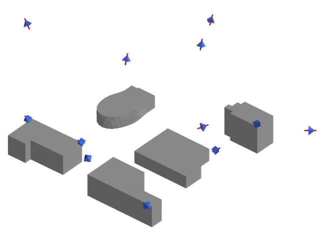
Figure 13: Convergence Rate



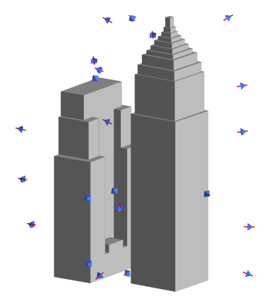
(a) Target building set 1



(b) Target building set 2



(c) Target building set 3



(d) Target building set 4

Figure 14: 3D visualization of the resultant viewpoints and the target buildings

viewpoints efficiently. The visibility information is encoded in the visibility matrix, and the problem is formulated as partial Set Covering Problem, which is then solved by Random-Key Genetic Algorithm and Greedy search methods. We have also demonstrated that the proposed method is able to find better planning solutions (fewer viewpoints and higher coverage ratio) compared to previous methods.

The future work will consider extending the view planning method to multiple UAVs systems with heterogeneous vision sensors, in order to further improve the efficiency in these inspection and surveillance applications.

Acknowledgment

Dr. Soji Yamakawa and Ved Vyas have provided invaluable help on the code and software development of this project. One of the authors, Wei Jing, has been supported by Agency for Science, Technology and Research (A*STAR), Singapore.

References

[1] W. Jing, J. Polden, W. Lin, K. Shimada, Sampling-based view planning for 3d visual coverage task with unmanned aerial vehicle, in: IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, 2016, pp. 1808–1815.

[2] A. Bircher, K. Alexis, M. Burri, P. Oettershagen, S. Omari, T. Mantel, R. Siegwart, Structural inspection path planning via iterative viewpoint resampling with application to aerial robotics, in: IEEE International Conference on Robotics and Automation, IEEE, 2015, pp. 6423–6430.

[3] K. Tarabanis, R. Y. Tsai, A. Kaul, Computing occlusion-free viewpoints, IEEE Transactions on Pattern Analysis and Machine Intelligence 18 (3) (1996) 279–292.

[4] S. Son, K. H. Lee, Automated scan plan generation using stl meshes for 3d stripe-type laser scanner, in: Computational Science and Its Applications-ICCSA 2003, Springer, 2003, pp. 741–750.

[5] G. H. Tarbox, S. N. Gottschlich, Planning for complete sensor coverage in inspection, Computer Vision and Image Understanding 61 (1) (1995) 84–111.

[6] W. R. Scott, G. Roth, J.-F. Rivest, View planning for automated three-dimensional object reconstruction and inspection, ACM Computing Surveys (CSUR) 35 (1) (2003) 64–96.

[7] S. Chen, Y. Li, N. M. Kwok, Active vision in robotic systems: A survey of recent developments, The International Journal of Robotics Research 30 (11) (2011) 1343–1377.

[8] M.-D. Dios, A. Ollero, et al., Automatic detection of windows thermal heat losses in buildings using uavs, in: Automation Congress, 2006. WAC'06. World, IEEE, 2006, pp. 1–6.

[9] I. Gentilini, Multi-Goal Path Optimization for Robotic Systems with Redundancy based on the Traveling Salesman Problem with Neighborhoods, Ph.D. thesis, Carnegie Mellon University (May 2012).

[10] U. Nilsson, P. Ogren, J. Thunberg, Optimal positioning of surveillance ugvs, in: IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, 2008, pp. 2539–2544.

[11] P. Cheng, J. Keller, V. Kumar, Time-optimal uav trajectory planning for 3d urban structure coverage, in: IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, 2008, pp. 2750–2757.

[12] L. Geng, Y. Zhang, J. Wang, J. Y. Fuh, S. Teo, Mission planning of autonomous uavs for urban surveillance with evolutionary algorithms, in: IEEE International Conference on Control and Automation, IEEE, 2013, pp. 828–833.

[13] K. A. Tarabanis, R. Y. Tsai, P. K. Allen, The mvp sensor planning system for robotic vision tasks, IEEE Transactions on Robotics and Automation 11 (1) (1995) 72–85.

[14] W. R. Scott, Model-based view planning, Machine Vision and Applications 20 (1) (2009) 47–69.

[15] K. A. Tarabanis, P. K. Allen, R. Y. Tsai, A survey of sensor planning in computer vision, IEEE Transactions on Robotics and Automation 11 (1) (1995) 86–104.

[16] E. Galceran, M. Carreras, A survey on coverage path planning for robotics, Robotics and Autonomous Systems 61 (12) (2013) 1258–1276.

[17] A. Mavrinac, X. Chen, Modeling coverage in camera networks: A survey, International journal of computer vision 101 (1) (2013) 205–226.

[18] E. Semsch, M. Jakob, D. Pavlicek, M. Pechoucek, Autonomous uav surveillance in complex urban environments, in: IEEE/WIC/ACM International Joint Conferences on Web Intelligence and Intelligent Agent Technologies., Vol. 2, IET, 2009, pp. 82–85.

[19] W. Jing, J. Polden, P. Y. Tao, W. Lin, K. Shimada, View planning for 3d shape reconstruction of buildings with unmanned aerial vehicles, in: International Conference on Control, Automation, Robotics and Vision, IEEE, 2016, pp. 1–6.

[20] B. Englot, F. S. Hover, Three-dimensional coverage planning for an underwater inspection robot, The International Journal of Robotics Research 32 (9-10) (2013) 1048–1073.

[21] S. Chen, Y. Li, Automatic sensor placement for model-based robot vision, IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics 34 (1) (2004) 393–408.

[22] R. M. Haralick, S. R. Sternberg, X. Zhuang, Image analysis using mathematical morphology, IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-9 (4) (1987) 532–550.

[23] M. Price, C. Stops, G. Butlin, A medial object toolkit for meshing and other applications, in: Proceedings of 4th International Meshing Roundtable, Citeseer, 1995, pp. 219–229.

[24] H. Blum, R. N. Nagel, Shape description using weighted symmetric axis features, Pattern recognition 10 (3) (1978) 167–180.

[25] S. A. Wilmarth, N. M. Amato, P. F. Stiller, Maprm: A probabilistic roadmap planner with sampling on the medial axis of the free space, in: IEEE International Conference on Robotics and Automation, Vol. 2, IEEE, 1999, pp. 1024–1031.

[26] H.-Y. C. Yeh, J. Denny, A. Lindsey, S. Thomas, N. M. Amato, Umaprm: Uniformly sampling the medial axis, in: IEEE International Conference on Robotics and Automation, IEEE, 2014, pp. 5798–5803.

[27] D. Reniers, J. J. van Wijk, A. Telea, Computing multiscale curve and surface skeletons of genus 0 shapes using a global importance measure, IEEE Transactions on Visualization and Computer Graphics 14 (2) (2008) 355–368.

[28] V. Boor, M. H. Overmars, A. F. van der Stappen, The gaussian sampling strategy for probabilistic roadmap planners, in: IEEE International Conference on Robotics and Automation, Vol. 2, IEEE, 1999, pp. 1018–1023.

[29] K. Shimada, D. C. Gossard, Bubble mesh: automated triangular meshing of non-manifold geometry by sphere packing, in: Proceedings of the third ACM symposium on Solid modeling and applications, ACM, 1995, pp. 409–419.

[30] T. Möller, B. Trumbore, Fast, minimum storage ray-triangle intersection, Journal of graphics tools 2 (1) (1997) 21–28.

[31] R. Hartley, A. Zisserman, Multiple view geometry in computer vision, Cambridge university press, 2003.

[32] V. Chvatal, A greedy heuristic for the set-covering problem, Mathematics of operations research 4 (3) (1979) 233–235.

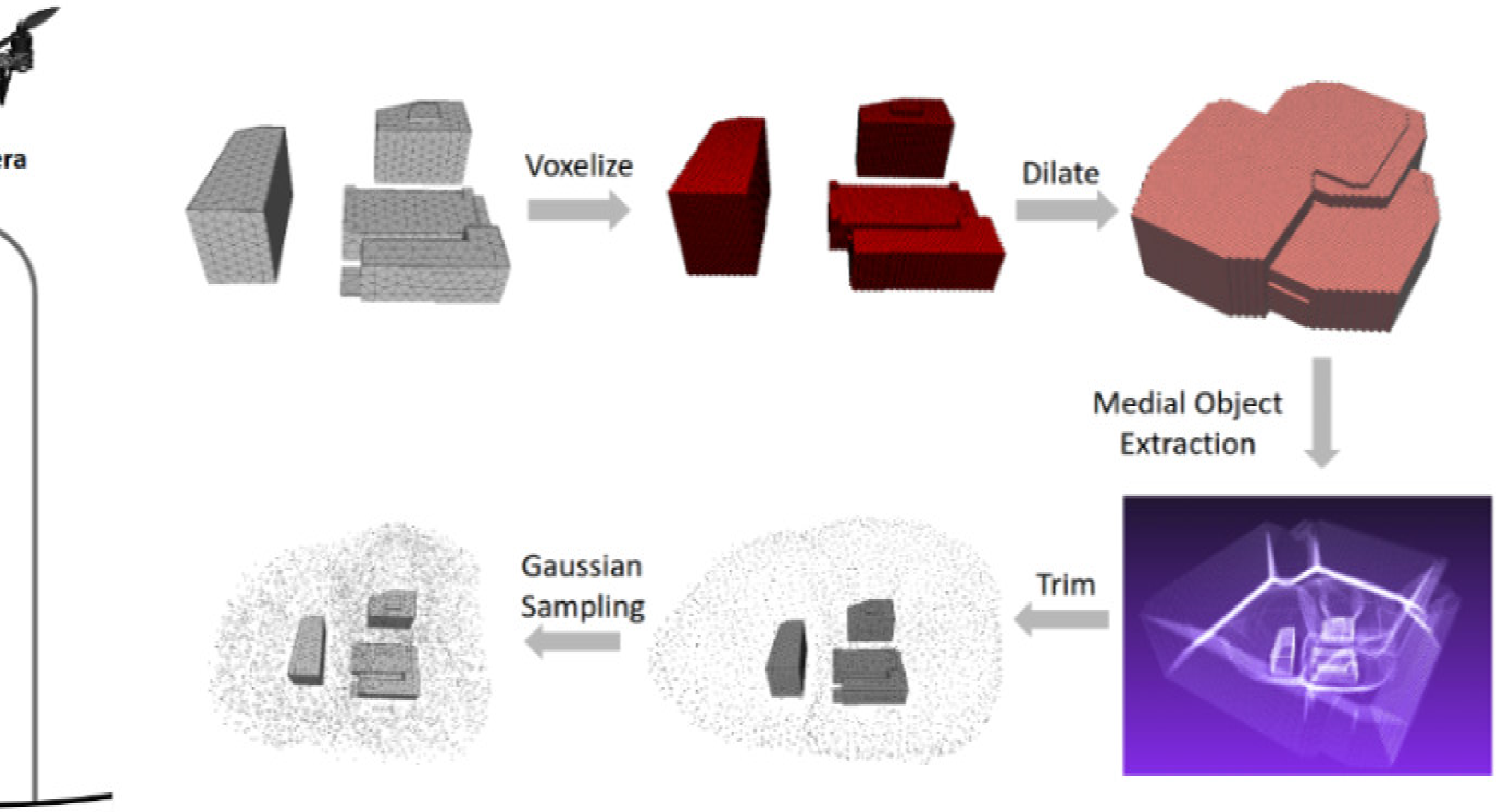
[33] Q.-S. Hua, Y. Wang, D. Yu, F. Lau, Dynamic programming based algorithms for set multicover and multiset multicover problems, Theoretical Computer Science 411 (26) (2010) 2467–2474.

[34] W.-C. Huang, C.-Y. Kao, J.-T. Horng, A genetic algorithm approach for set covering problems, in: Proceedings of the First IEEE Conference on Evolutionary Computation, IEEE World Congress on Computational Intelligence., IEEE, 1994, pp. 569–574.

[35] J. C. Bean, Genetic algorithms and random keys for sequencing and optimization, ORSA journal on computing 6 (2) (1994) 154–160.

[36] J. F. Gonçalves, M. G. Resende, Biased random-key genetic algorithms for combinatorial optimization, Journal of Heuristics 17 (5) (2011) 487–525.

[37] Mathworks, MATLab Genetic Algorithm, The MathWorks Inc., 2017. URL http://www.mathworks.com/help/gads/genetic-algorithm.html



Candidate viewpoints
generation

- A two-step “generate-test” view planning framework is proposed for model-based view planning problems.
- Voxel dilation and Medial Objects are used along with Gaussian sampling method to generate candidate viewpoints.
- Random-Key Genetic Algorithm and Greedy search method are combined to solve the formulated partial Set Covering Problem
- The proposed method is benchmarked with two previous methods under different conditions, and it outperforms those previous methods by requiring less viewpoints and achieves better coverage