remote sensing

MDPI

Article

Exploration for Object Mapping Guided by Environmental Semantics Using UAVs

Reem Ashour [1,*], Tarek Taha [2], Jorge Manuel Miranda Dias [1], Lakmal Seneviratne [1] and Nawaf Almoosa [3]

[1] Khalifa University Center for Autonomous Robotic Systems (KUCARS), Khalifa University of Science and Technology (KU), 127788 Abu Dhabi, UAE; jorge.dias@ku.ac.ae (J.M.M.D.); lakmal.seneviratne@ku.ac.ae (L.S.)
[2] Algorythma's Autonomous Aerial Lab, 112230 Abu Dhabi, UAE; tarek@tarektaha.com
[3] EBTIC, Khalifa University of Science and Technology (KU), 127788 Abu Dhabi, UAE; nawaf.almoosa@ku.ac.ae
* Correspondence: reem.ashour@ku.ac.ae

Received: 9 February 2020; Accepted: 3 March 2020; Published: 10 March 2020

Abstract: This paper presents a strategy to autonomously explore unknown indoor environments, focusing on 3D mapping of the environment and performing grid level semantic labeling to identify all available objects. Unlike conventional exploration techniques that utilize geometric heuristics and information gain theory on an occupancy grid map, the work presented in this paper considers semantic information, such as the class of objects, in order to gear the exploration towards environmental segmentation and object labeling. The proposed approach utilizes deep learning to map 2D semantically segmented images into 3D semantic point clouds that encapsulate both occupancy and semantic annotations. A next-best-view exploration algorithm is employed to iteratively explore and label all the objects in the environment using a novel utility function that balances exploration and semantic object labeling. The proposed strategy was evaluated in a realistically simulated indoor environment, and results were benchmarked against other exploration strategies.

Keywords: semantic exploration and mapping; utility function; cost function; semantic mapping; semantic exploration; next-best-view

1. Introduction

The growth in aerial robotics has led to their ubiquitous presence in various fields—urban search and rescue (USAR) [1–3], infrastructure inspection [4], surveillance [5], etc. Some recent research has focused on USAR activities performed by unmanned aerial vehicles (UAVs) to assist rescue teams by providing vital information on time-sensitive situations without endangering human lives. The introduction of unmanned aerial vehicles (UAVs) in USAR environments is beneficial when rapid responses are required to assist first responders with locating victims, survivors, and danger sources in the environment. UAVs' speed, agility, and ability to navigate in hazardous environments that contain rubble and obstacles make them an ideal platform for deployment in USAR environments. UAVs with autonomous exploration, mapping, and navigation capabilities can provide rescuers with valuable data, such as 2D and 3D maps, victims' locations, and localized danger sources, in order to improve their awareness of the situation and respond appropriately.

Maps can provide a variety of information about the environment; they can reflect the structure of the environment, connections, and knowledge sense. Therefore, providing a rich/informative map for the first responders in USAR with the right representation is an active topic in the USAR robotics field.

Robot pose and sensor data are used to build a globally consistent map in which path planning and exploration become feasible. In the literature, maps are classified by the way they represent information into three types [6,7]: topological [8], metric [8,9], and semantic maps [10–13]. The majority of the existing mapping approaches aim to construct a global consistence metric map of robots operating in the environment [6–8,14]. Metric maps do not encapsulate the characteristics that are compatible with human conception. Therefore, semantic maps are preferable to be used for high-level objectives, such as detecting and identifying the locations of different objects, path planning, and increasing environmental awareness.

Autonomous exploration could be performed to minimize the effort required by the first responders in a USAR environment to search for victims and build awareness maps. Many approaches have been heavily used in the literature to perform autonomous exploration, such as frontier-based exploration [15] and information-theory-based exploration [16]. The concept of "next-best-view" (NBV) [17] was extensively used to guide the next best exploration action, whether using the frontier-based or the information-theory-based concepts. In order to identify the best next exploration action, a cost function (also called utility function) is used to evaluate various views to identify the one that maximizes the objectives of the exploration.

Both frontier-based exploration and information-theory-based exploration approaches of unknown indoor environments use metric (i.e., 2D/3D occupancy) maps without considering the high-level abstract of contextual information provided by onboard sensors, such as the class of objects in occupied voxels. Additionally, both approaches do not filter or classify which information is of merit to the application-specific environment, which is responsible for reducing the time of finding objects of interest or labeling all the objects in the environment.

In this work, an efficient strategy for autonomous, semantically-aware exploration of object labeling and 3D mapping for unknown indoor is proposed. The proposed approach utilizes semantic information encapsulated in the proposed 3D, semantically-aware map for object localization and labeling using a UAV with an onboard RGBD camera. New utility functions are proposed to direct the robot towards the objects in the environment. The results show that the proposed strategy is capable of exploring unknown environments and label objects effectively. Our contributions can be summarized as follows:

- A 3D, semantically-aware, colored annotated mapping method that uses a deep learning model to semantically segment the objects in 2D images and project the annotated objects in a 3D, semantically-aware occupancy map. This includes a novel data structure that extends the volumetric occuapancy grid representations to include more semantic specific information.
- Development of a multi-objective utility function that encapsulates the quantified information generated from the semantic map and volumetric map to allow the robot to explore the unknown space and to direct the robot to visit the objects and label them effectively.
- An overall exploration strategy that utilizes rapidly exploring random tree (RRT) for viewpoint sampling, and the next-best-view approach with the proposed semantic utility functions to iteratively explore the unknown environment and label all the objects there.

The remainder of the paper is structured as follows. Section 2 provides an overview of the related work. Section 3 details the methodology presented in this work, wherein Section 3.1 illustrates the proposed strategy and describes the main components, Section 3.2 represents the semantic mapping strategy employed in this work, and Section 3.3 presents the proposed exploration approach and its components. Section 4 shows our experiment setup and preliminary results of our simulation experiments, and finally, Section 5 summaries our contributions and provides recommendations for future work.

2. Related Work

Robots deployed in search and rescue missions should possess basic yet very important skills, such as object detection, 3D mapping, and indoor exploration. The significant impact of the unmanned aerial vehicles (UAVs) in various robotics fields made them a promising solution for urban search and rescue tasks. For victim detection, the UAV could utilize the onboard sensors and the onboard intelligence to accurately detect and locate victims in disaster scenes in order to increase the likelihood of finding victims [18–20]. For mapping and 3D reconstruction, the UAV should be capable of reconstructing the scene in 3D and locating the contextual and rich information that can be used by the rescuers' team. For exploration, the UAV should be able select the exploration positions in order to facilitate both the victim detection and 3D reconstruction tasks. In this section, the related work for the object detection, semantic mapping, and exploration is provided.

2.1. Object Detection and Classification

Victim detection is the most critical task in search and rescue missions, wherein victims should be identified as fast as possible for the first responders to intervene and rescue them. Object detection task is concerned with discovering instances of semantic objects that belong to a particular class, such as human, table, or chair. In the literature, a variety of mechanisms are used to detect objects either by using a single sensor or fusing multiple sensors [20].

Visual approaches for object detection is the most reliable resource used to find objects in scenes. The visual object detection is extensively performed using machine learning methods and currently through deep learning techniques. Both techniques depend on the visual information provided by an individual image frame. In the context of machine learning approaches, a feature extraction method, such as scale-invariant feature transform (SIFT) [21] or histogram of oriented gradients (HOG) features [22], is used to define the features. Then, a machine learning model, such as supportive-vector-machine (SVM) [23], is employed for object classification. In deep learning, the neural networks are capable of detecting objects without the feature extraction process. There are different types of object-detection-based deep learning model techniques depending on the source of information captured from the onboard sensor, such as 2D images and 3D point cloud.

Utilizing the various information captured from the onboard sensors such as 2D images and 3D point cloud, different deep learning models are designed to detect objects. Commonly, convolutional neural networks (CNNs) are used to detect and identify objects from 2D images. The most commonly used deep learning models for object detection from 2D images are the single shot multiBox detector (SSD) [24], and the you only look once (YOLO) [25–27]. These methods create a bounding box that surrounds the classified objects in the 2D image. Nonetheless, object detection and classification can be performed using the deep learning models that apply semantic segmentation for 2D images, such as PSPNet [28]. However, using depth images, object detection and classification is performed using the deep learning networks that segment point cloud and classify the segments to the different classes, such as PointNet [29], PointNet++ [30], and Frustum pointnets [31].

2.2. Semantic Mapping

A semantic map is an augmented representation of the robot's environment that encapsulates characteristics compatible with human conception. Exploration and path planning algorithms benefit from the additional information that the semantic maps provide about the surrounding environment. Most of existing mapping strategies intend to build a global consistency metric map of robots operating in the environment [6–8,14,32]. In USAR, the robot should be capable of guiding the first responders to the victim's location safely. Therefore, the robots should acquire more advanced skills than the metric mapping; the robot should understand the surrounding environment. A solution can be found by the semantic mapping, which is a qualitative representation of the robot's surrounding environment to augment the exploration and task planning. In the literature, since contemporary robots use the metric maps to explore the environment, most of the work on semantic maps uses the metric maps as a basis for the semantic map and builds on top of semantic attributes; see [33,34].

2.3. Exploration

Two of the most commonly used autonomous exploration approaches are frontier-based approaches and information-theory-based approaches. In **frontier-based exploration** approaches, the robot navigates to the boundaries between free space and unexplored space by evaluating the cost function to maximize map coverage [15]. Utility functions, such as path cost and expected information gain, are usually applied to evaluate next candidate position (i.e., frontier) to visit. However, in information-theory-based approaches, the robot navigates to candidate position (i.e., viewpoint sample) to maximize information gain using probabilistic map representation [16]. Therefore, a good information gain prediction algorithm plays a vital role in both methods. During exploration, different types of maps can be created to either reconstruct the 3D scene or to present the scene with varying types of information depending on their semantics utilizing the obtained data from the onboard sensors [6]. Advanced variant extensions of this frontier-based algorithm were developed in [35–37].

The commonly used exploration techniques of unknown indoor environments use metric (i.e., 2D/3D occupancy) maps without considering the high-level abstract of information provided by onboard sensors, such as the class of objects in occupied voxels. These methods may work locally on simple environments but they are unlikely to function well in cluttered and dynamic environments. These drawbacks motivate the need for a new efficient exploration strategy that evaluates the contextual information from the environment. Fortunately, current technological advancements in hardware and software algorithms (such as deep learning) have provided a new research direction by encapsulating semantic in occupancy maps that allows efficient object labeling and localization.

Recently, autonomous **semantic-based exploration** applied in [33,34] has used the contextual information encapsulated in semantic maps to enhance the exploration process by maximizing the use of available resources. T. Dang et al. [34] proposed an autonomous exploration model that explores an unknown environment and simultaneously searches for objects of interest. The proposed method utilizes a deep learning model that classifies the objects from 2D images and projects them into a 3D semantic occupied map. Each object of interest occupies a voxel in the 3D map with a specific color. The exploration process uses the 3D semantic occupied map to search for objects of interest and to explore the unknown environment. T. Dang et al. in [33] employed the visual saliency model to build an annotated map using visual importance incrementally. The exploration planner simultaneously optimizes the exploration of unknown spaces and directs the robot towards the salient objects.

In contrast to these methods, we firstly propose creating a new, 3D, semantically-aware occupancy map structure that carries the occupancy information and the semantic annotations to be utilized in the autonomous exploration task. A deep learning model is utilized to semantically segment the objects from 2D images and project them into a 3D map where each object is labeled with a unique color, as illustrated in Section 3.2. Secondly, we propose a new multi-objective utility function that uses the semantic information to direct the robot towards the objects in the environment and label them as shown in Section 3.3. Our work is different from the studies in the literature; instead of using only an occupancy map for autonomous exploration, we utilize the contextual information from the 3D semantic map to explore the environment and label all the objects. Besides, the semantic map utilizes a deep learning model to semantically segment the objects in 2D, which allows providing an accurate projection of objects in the 3D semantic map. Additionally, our proposed utility function encapsulates the contextual information by either engaging the confidence value of object belonging to a particular class or the number of visits for objects of interest, thereby it facilitates the labeling purpose.

3. Materials and Methods

3.1. Proposed Approach

In this work, a bounded exploration area $V \in R^3$ is initially unknown and has to be explored. Our exploration strategy attempts to create a 3D map M and find a collision-free path σ starting from an initial position ζ_{init} that leads to the following: (a) Identify the free V_{free} and occupied V_{occ} parts of the whole exploration area and (b) maximize the information gathering to label objects in the environment using two sources of information: (1) the depth information captured by the robot sensor and (2) the semantic information obtained from a deep learning model.

Figure 1 shows the proposed semantically-aware exploration of object labeling and 3D mapping system architecture. The proposed system architecture consists of two main modules: (1) mapping and information quantification, and (2) autonomous exploration. The proposed approach aims to provide the first responders in USAR scenarios a rich environmental representation.

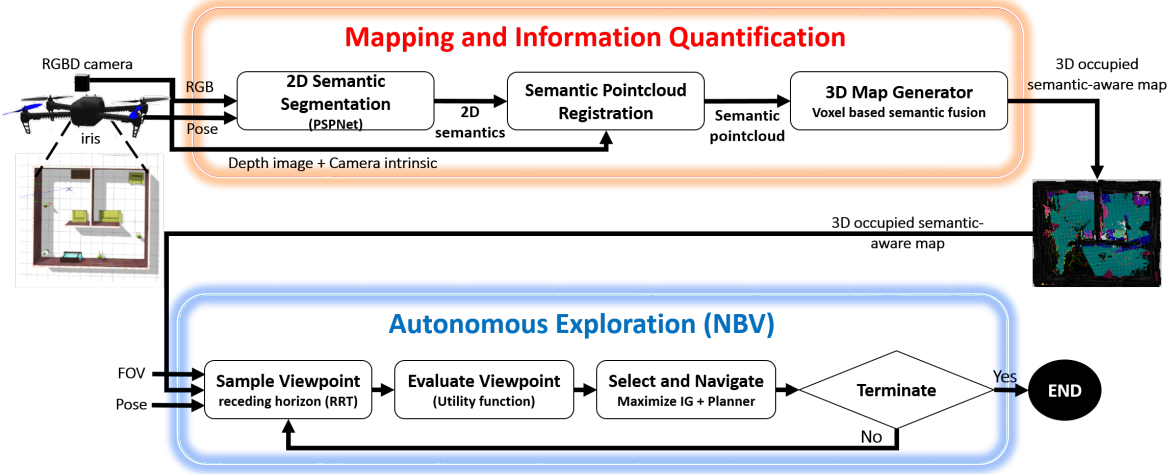

Mapping and Information Quantification
RGBD camera
iris
RGB
Pose
2D Semantic Segmentation (PSPNet)
2D semantics
Semantic Pointcloud Registration
Semantic pointcloud
3D Map Generator
Voxel based semantic fusion
3D occupied semantic-aware map
Depth image + Camera intrinsic
3D occupied semantic-aware map
Autonomous Exploration (NBV)
FOV
Pose
Sample Viewpoint
receding horizon (RRT)
Evaluate Viewpoint
(Utility function)
Select and Navigate
Maximize IG + Planner
Terminate
Yes
No
END

Figure 1. Proposed semantically-aware exploration of object labeling, and the 3D mapping system architecture.

In the first module, the 3D semantic-occupied map is generated by using a deep learning model to segment the objects in 2D images into different labels. To perform this step, semantic segmentation is used to perform object classification and segmentation in pixels level and assign each pixel to a particular class. The point clouds corresponding to the image pixels are annotated using the output from the deep learning model. The annotated point clouds are used to create a 3D semantic occupancy map. The 3D semantic-occupied map is divided into small cubical voxels $m \in M$ with edge length *res* that refers to the map resolution. Each voxel stores three types of information, occupancy, color, and confidence values, which correspond to the voxel's vacancy, class, and certainty, respectively.

Initially, all the voxels are assigned to be unknown. After the initial data gathering from the onboard sensors, the 3D semantic occupancy map is updated to contain $V_{free} \in V$, $V_{occ} \in V$, and the updated information in each voxel about the number of visits, confidence values, and class type. Feasible paths σ are subject to the limited field of view (FOV) and effective sensing distance constraints.

In the second module, the autonomous exploration module uses the contextual information obtained from the mapping stage. In this work, we adopted the NBV algorithm to determine the next best view that fulfills the ultimate goal to label all objects in the environment eventually. The exploration module involves three stages: (i) viewpoint sampling by employing a receding horizon, (ii) viewpoint evaluation (iii) navigation, and termination.

Two new multi-objective utility functions are proposed to evaluate each candidate viewpoint sample in terms of two criteria: (i) information gain obtained from exploring new space that has not been mapped yet; (ii) the information gain obtained from semantically labeled objects with either lower confidence or visits values for the object of interest to direct the robot to label the objects in the unknown environment accurately.

The candidate that maximizes the gain is selected to be the next best view, and the robot moves to that position. The whole process is repeated in a receding horizon fashion. Paths are only planned in the free space V_{Free}. Within the algorithm's implementation, the robot configuration is given as $P = [x, y, z, yaw]$, where roll and pitch are considered near-zero. In this work, the following assumptions were made; (i) the robot's position was assumed to fully known since the simulation experiment was performed on the Gazebo simulator, and the position was obtained directly through the simulated models of the UAV (robot localization is out of the scope of this research); (ii) the environment is unknown; and (iii) the evaluation of utility functions for the exploration process is performed on a voxel scale rather than object scale.

3.2. Mapping and Information Quantification

The procedure for the mapping and information quantification is provided in Figure 1. It involves three stages: scene understanding, where semantic segments are generated for objects found in a 2D image frame; annotated point cloud generation, where the point cloud captured from the depth camera is annotated to the corresponding class from the deep learning model output; and the 3D, semantically-aware mapping, which registers the detection annotated point cloud to 3D occupancy grid map. The stages are explained in detail in the following sections.

3.2.1. 2D-Image-Based Semantic Segmentation and Semantic Fusion

- **Semantic Segmentation**

The deep neural network pyramid scene parsing network (PSPNet) based on semantic segmentation [28] is employed to provide semantic segments for the different objects in 2D images. The network architecture is shown in Figure 2. PSPNet operates in five steps as follows: (i) create a feature map using resnet, (ii) apply pyramid pooling on the created feature map to produce feature maps at varying resolutions, (iii) perform convolution on each pooled feature map, (iv) create a final feature map by upsampling the feature maps and concatenate them, and (v) generate a class score map by employing final convolutions.

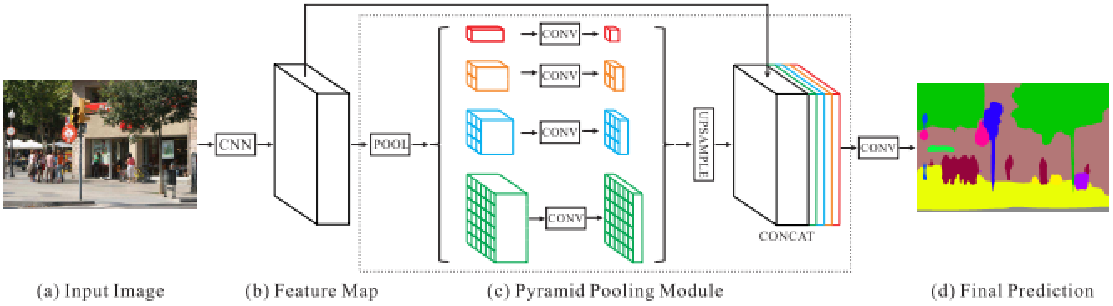
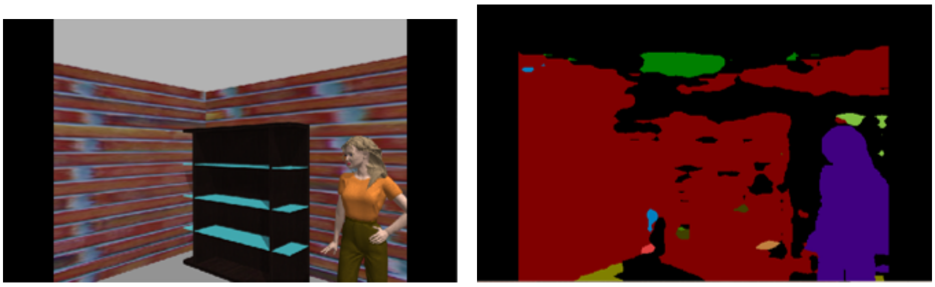


Figure 2. Pyramid scene parsing network (PSPNet). Extracted from [28].

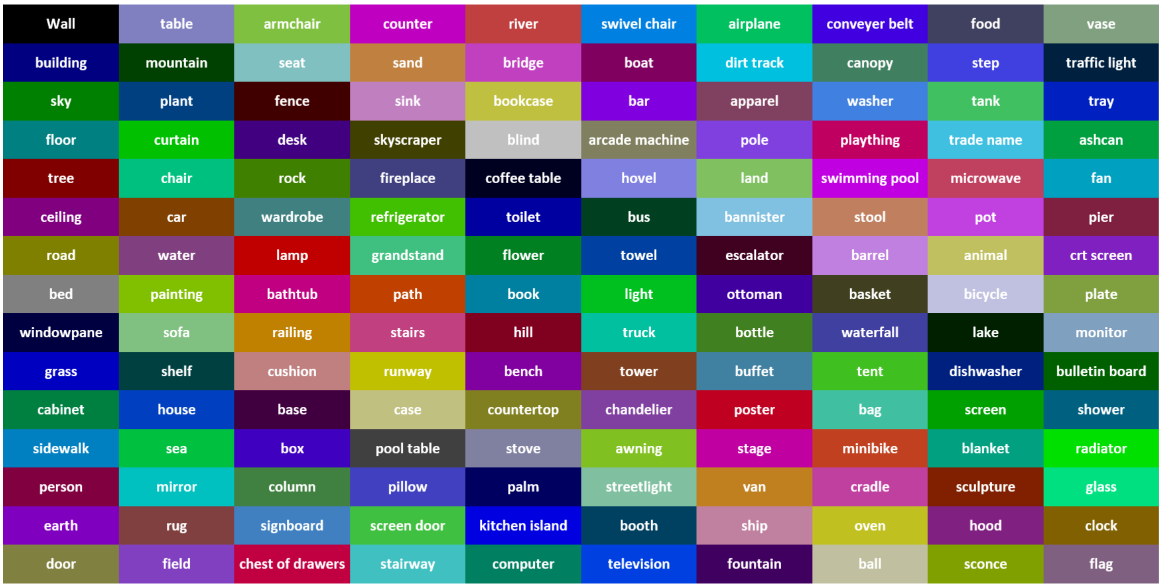
The PSPNet model is trained [28] on two datasets, the SUNRGB [38] and ADE20K [39]. The ADE20K dataset includes a large vocabulary of 150 classes from both indoor and outdoor objects. However, the SUNRGBD data set contains 38 semantic classes for indoor scenes. The fact that the ADE20K dataset contains a larger number of objects made it a preferable choice to segment an unknown environment with a diversity of objects. Examples of 2D images with semantic segments are shown in Figure 3a,b respectively. The RGB images are obtained using the simulated sensor in the Gazebo simulator that mimics the characteristics of an actual camera.

In this work, only ADE20K dataset is used and the semantic labels of ADE20K dataset are shown in Figure 4.



(**a**) RGB image/color textured. (**b**) Semantic segmentation for a 2D image.

Figure 3. RGB image and semantic segment image.



Wall	table	armchair	counter	river	swivel chair	airplane	conveyer belt	food	vase
building	mountain	seat	sand	bridge	boat	dirt track	canopy	step	traffic light
sky	plant	fence	sink	bookcase	bar	apparel	washer	tank	tray
floor	curtain	desk	skyscraper	blind	arcade machine	pole	plaything	trade name	ashcan
tree	chair	rock	fireplace	coffee table	hovel	land	swimming pool	microwave	fan
ceiling	car	wardrobe	refrigerator	toilet	bus	bannister	stool	pot	pier
road	water	lamp	grandstand	flower	towel	escalator	barrel	animal	crt screen
bed	painting	bathtub	path	book	light	ottoman	basket	bicycle	plate
windowpane	sofa	railing	stairs	hill	truck	bottle	waterfall	lake	monitor
grass	shelf	cushion	runway	bench	tower	buffet	tent	dishwasher	bulletin board
cabinet	house	base	case	countertop	chandelier	poster	bag	screen	shower
sidewalk	sea	box	pool table	stove	awning	stage	minibike	blanket	radiator
person	mirror	column	pillow	palm	streetlight	van	cradle	sculpture	glass
earth	rug	signboard	screen door	kitchen island	booth	ship	oven	hood	clock
door	field	chest of drawers	stairway	computer	television	fountain	ball	sconce	flag

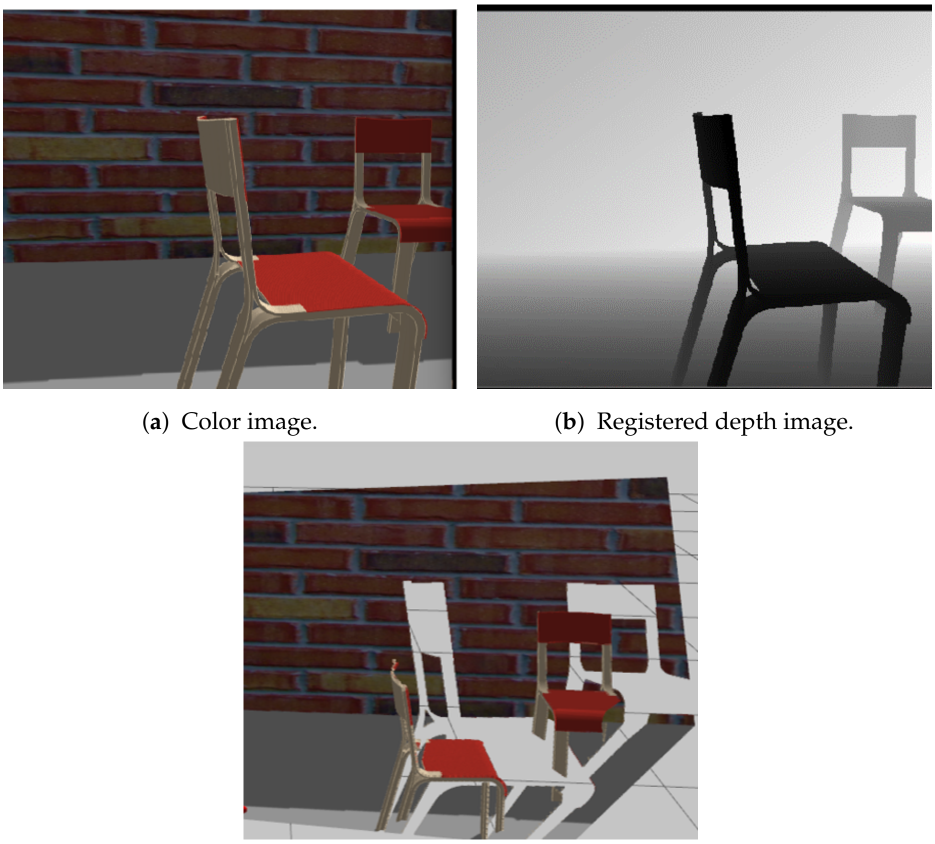
Figure 4. Semantic labels of ADE20K data set in BGR format.

- **Semantic Fusion**

  Semantic information encapsulated in the point cloud is fused using the "max fusion" approach [40]. In the max fusion approach, the semantic color that corresponds to the highest confidence value generated from the CNN model is included in the generated point cloud. In addition, the same value of the confidence and the label color are saved in the corresponding voxel of octomap.

3.2.2. Annotated Point Cloud Generation

The point cloud is a collection of points in an unordered structure where each point contains a coordinate in a specific reference frame. To generate a point cloud, firstly, the depth image is registered with the same reference frame that the color image is registered in, which is usually the camera frame. After that, the pixels are transformed from the camera frame to the real world frame using the image position, its depth, and the intrinsic camera parameters to form the point cloud. An example of the point cloud generation process is illustrated in Figure 5. The point-clouds are textured color. Therefore, the colors of the point clouds have the same colors as the objects presented in the image frame, as shown in Figure 5c. An illustration of the point cloud data structure when storing semantic information using max fusion is shown in Figure 6.

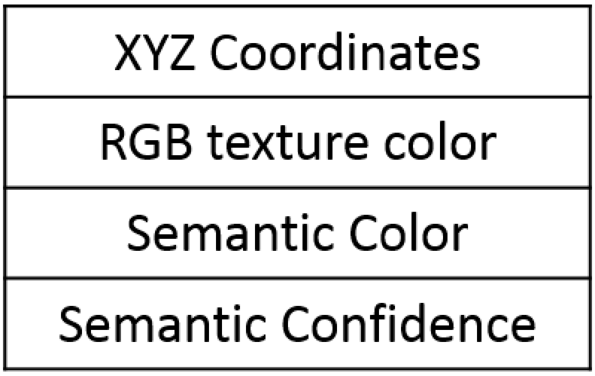


(**a**) Color image.

(**b**) Registered depth image.

(**c**) Generated point cloud.

Figure 5. Point cloud generation.



XYZ Coordinates

RGB texture color

Semantic Color

Semantic Confidence

Figure 6. Max fusion.

3.2.3. 3D Semantic Map Generation

To improve the semantic exploration strategies, the object classes detected and localized per depth and color sensor input are used to obtain a 3D semantic-occupied annotated map representation of the environment.

The proposed 3D occupied semantic map structure is based on an occupancy grid map called octomap [41] which uses octree structure. An octree is a hierarchical data structure that divides the spaces in cubes of the same size called voxels. Each division is then recursively subdivided into eight sub-volumes until a given minimum voxel size is reached, as shown in Figure 7. The resolution determines the voxel size. The map $M = \{m_1, m_2, ..., m_i\}$ consists of the cubical element of the same size $m_i \leftarrow$ voxels for index i. The voxels in the traditional occupancy maps can only hold volumetric information. However, in addition to the volumetric information, the proposed semantic occupancy map adds the semantic information to the voxels. As a result, each voxel m_i in the proposed map holds the volumetric information and semantic information. The semantic information includes the semantic color, confidence value, and number of visits. The confidence value is the value for the highest confidence generated from the CNN.

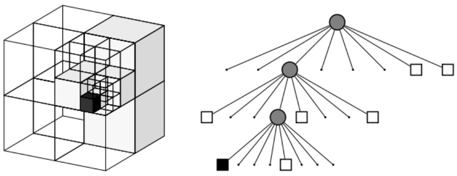


Figure 7. Right: an octree example where the free voxels are shaded white and occupied voxels are shaded black. Left: volumetric model that shows the corresponding tree representation on the right [41].

Each voxel m_i holds the following information:

1. Position (x, y, z);
2. Probability of being occupied (value) P_o;
3. Semantic color (r, g, b);
4. Confidence value $c(x)$ from class type (cs);
5. Number of visits.

Initially, all voxels are assigned to be unknown where $P_o = 0.5$, colored in white, and given confidence value 0. At every point cloud view, the input data to the mapping and exploration module are 2D image and 3D point cloud point clouds that consist of the positional and the semantic $(x, y, z, r, g, b, confidence)$ information. The color indicates a certain class type and not the texture color. The point clouds are used to build the voxels in the semantic-occupied map. This information is fed back into the map, and the corresponding position and semantic color are assigned to the voxels located within the camera FOV. The occupancy value $P_o(m_i) \in [0 : 1]$, which represents the probability of being occupied is assigned to the observed voxels using the volumetric mapping. Simultaneously, the class type is assigned to the voxel using semantic color indication. The visualized constructed map is colored according to the object class.

3.3. Semantically-Aware Exploration

The proposed exploration strategy provides the robot with the ability to explore unknown environments, while simultaneously optimizing information gathering and directing the robot to label objects in the environment. To enable this functionality, two new multi-objective utility functions are proposed to account for the semantic information (confidence or number of visits) that each voxel encapsulates. The proposed system used information from the semantic-occupied map to evaluate the next best exploration action. A technique used to explore an unknown environment is the next-best view (NBV) approach. The main steps in the exploration task are: (A) viewpoint sampling, (B) viewpoint evaluation, (C) navigating toward the selected viewpoint, and (D) termination. The exploration process is summarized in Figure 8. At the beginning of the exploration process, a robot uses the onboard sensors to observe the scene and produce a set of viewpoints candidates (also known as states or poses). The accepted viewpoint candidates are then evaluated using a utility function (also known as reward, cost, or heuristic function). The viewpoint which maximizes the utility function is selected as the next goal and is called "next-best view". The exploration process is repeated until a termination condition is met, indicating the end of the exploration. Algorithm 1 is used as a path planning procedure adapted from [42].

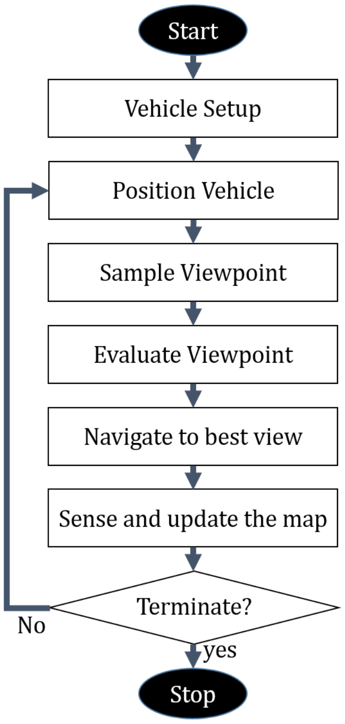


Figure 8. General components of the NBV method.

Algorithm 1 Proposed planner—iterative step.

```
$\xi_0$ current robot configuration
Initialize $T$ with $\xi_0$
$g_{best} \leftarrow 0$
$n_{best} \leftarrow n_0(\xi_0)$
$N_T \leftarrow$ Number of initial Nodes in $T$
while $N_T < N_{max}$ or $g_{best} = 0$ do
    Incrementally build $T$ by adding $n_{new}(\xi_{new})$
    $N_T \leftarrow N_T + 1$
    if InformationGain($n_{new}$) $> g_{best}$ then
        $g_{best} \leftarrow$ InformationGain($n_{new}$)
        $n_{best} \leftarrow n_{new}$
    end if
    if $N_T > N_{TOL}$ then
        Terminate
    end if
        $\sigma_{RH}, n_{RH}, \xi_{RH} \leftarrow$ ExtractBestSegment($n_{bes}$)
end while
return $\sigma_{RH}$
```

3.3.1. Viewpoint Sampling

The viewpoint samples are the possible positions the robot may visit as a next step. Taking all the possible views in consideration is computationally intractable and expensive [43,44]. Hence, a subset of possible views is selected for the evaluation step based on the information it can provide. Different sampling strategies are used, such as (i) regular grid sampling [45], (ii) frontier sampling [46], and (iii) incremental sampling (i.e., rapidly-exploring random trees (RRT)) [47].

The incremental approaches are different from the regular grid sampling and frontier sampling by selecting a series of points randomly in a tree-like manner instead of multiple single views for evaluation. These trees can be built by rapidly-exploring random trees (RRT) [48] or their variant RRT* [47]. The tree is expanded throughout all the exploration space, and each branch forms a group of random branches. RRT is extensively used due to its ability to work in higher dimension spaces, and it ensures probabilistic completeness. Each vertex in the search space has a Voronoi region that is calculated by the RRT, where the randomization guarantees a biased search towards vertices with more significant Voronoi regions.

Each point in the branch is evaluated using a utility function, and the branch which maximizes the utility function is chosen. Although the evaluation is performed for the whole branch, and the best branch is selected for execution, Bircher et al. [42] executes only the first edge of the selected branch. The rest of the branch is utilized for new tree initialization, maintaining the original path while allowing for new paths to grow as the map is updated. The rapidly-exploring random trees and the viewpoint sampling approach provided in [42] are adapted in this research, as shown in Figure 9.

In each iteration, a random geometric tree $\mathbf{T}$ of finite depth with maximum edge length of l is sampled in the known free space of robot configurations and incrementally built from an initial configuration x_{i0}. All branches of this tree are evaluated using a utility function. When the best branch is extracted, the first viewpoint ξ_{RH} that corresponds to the vertex n_{RH} from the path σ_{RH} is selected and the robot moves to that point.

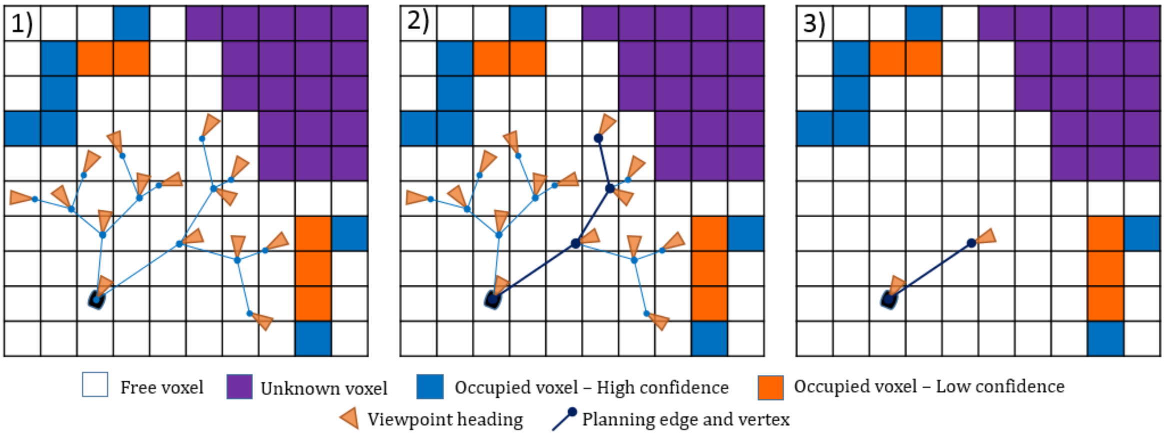


Figure 9. Overview of the functional concept of the proposed semantically exploration planner. At every iteration, a random tree of finite depth is sampled in the known free space of robot configurations. All branches of this tree are evaluated in terms of a combined information gain related to exploration of unknown areas and semantic-information-based class confidence value as represented in the occupancy map. The best branch is identified, and the step to its first viewpoint is conducted by the robot. Subsequently, the whole procedure is repeated in a receding horizon fashion.

The viewpoint or edges are added to the tree and are accepted for evaluation only if they were inside the exploration area, and the direct line of site from its parent does not cross occupied voxels, or in other terms, the path is collision free.

3.3.2. Viewpoint Evaluation

The utility function is used to select the best candidate viewpoint for the robot to visit. Many factors impact the viewpoint evaluation, such as the cost of moving to a new point in terms of distance, time, and power consumed, as well as the expected information gain (IG) the viewpoint can provide. The proposed utility function utilizes both volumetric and semantic information that a single voxel holds. The gain is calculated for each branch as the accumulated gain for the single nodes in the branch.

At each viewpoint ξ_{RH} sample, the sensor generates a set of rays R where the rays end if it incident a physical surface or reaches the limit of the map. For a single view v within a set of views V, the 3D points from the sensor are annotated and projected into the map. The projection is carried through the ray casting, resulting in a set of rays R_v cast for every view. Each ray r traverses the map, the volumetric and semantic information are accumulated within a set of visited voxels X. The predicted information gain G_v for a single view is the cumulative volumetric information and semantic information collected along the rays cast from v, such that:

$$G_v = \sum_{\forall r \in R} \sum_{\forall x \in X} \alpha I_{vol}(x) + \beta I_{sem}(x) \tag{1}$$

where $I_{vol}(x)$ is the volumetric information, $I_{sem}(x)$ is the semantic information that a single voxel encapsulates, and α and β are the weights for alternating between the volumetric and semantic gains. The formulation of the $I_{vol}(x)$ and $I_{sem}(x)$ defines the behavior of the system.

For a single voxel, the uncertainty that encodes occupancy probability is defined as voxel entropy, as in Equation (2).

$$H(x) = P_o \ln(P_o) + (\bar{P_o}) \ln(\bar{P_o}) \tag{2}$$

where $P_o \in [0:1]$ donates the probability of voxel x being occupied and $\bar{P_o}$ is the complement probability of P_0; i.e., $\bar{P_o} = 1 - P_o$. A voxel where no information is provided about it has $P_o = 0.5$ and has the highest entropy $H(x) = 1$ *Shannon*. Numerous research in the literature considered the volumetric information to formulate utility functions for different exploration purposes, such as volumetric gain [42], pure entropy [49], average entropy [50], occlusion aware [49], and unobserved voxels [49].

The volumetric gain and pure entropy utility functions tend to reduce the map entropy by directing the robot towards the views that contain a more significant amount of unknown voxels. However, information gain obtained via average entropy favors the views traverse a fewer number of voxels with higher entropy. In addition, the occlusion aware utility function tends to visit views with more unknown voxels near to the candidate view.

Utility functions in the literature count for volumetric,= or visibility likelihood of the voxels and neglect the semantic information. We propose two multi-objective functions that consider both volumetric and semantic information encapsulated in each voxel. In this research, the volumetric information is adapted from [42], where the $I_{vol}(x)$ is defined as the accumulative number of the unknown visible voxel in the candidate viewpoint. Let $I_{vis}(x)$ be the indicator that the voxel is visible. A voxel is visible if the line of sight from the robot to the voxel does not contain an occupied voxel. Hence, $I_{vis}(x)$ is defined as:

$$I_{vis}(x) = \begin{cases} 1 & x \in visible(m_{unknonwn}) \\ 0 & otherwise \end{cases} \tag{3}$$

However, we propose two different multi-objective utility functions, semantic visible voxel (SVV) and semantic visited object Of interest visible voxel (SVOI), with two different semantic information formulations, consequently.

- **Semantic Visible Voxel (SVV)**

  The proposed SVV multi-objective function tends to direct the robot towards the views that contain known occupied voxels with a small confidence value to label all the objects in the scene. The confidence value is obtained from CNN, and it is the maximum confidence value for the corresponding class out of 150 objects. Let $I_c(x)$ be the indicator if the semantic confidence value assigned to a voxel is less than a predefined threshold:

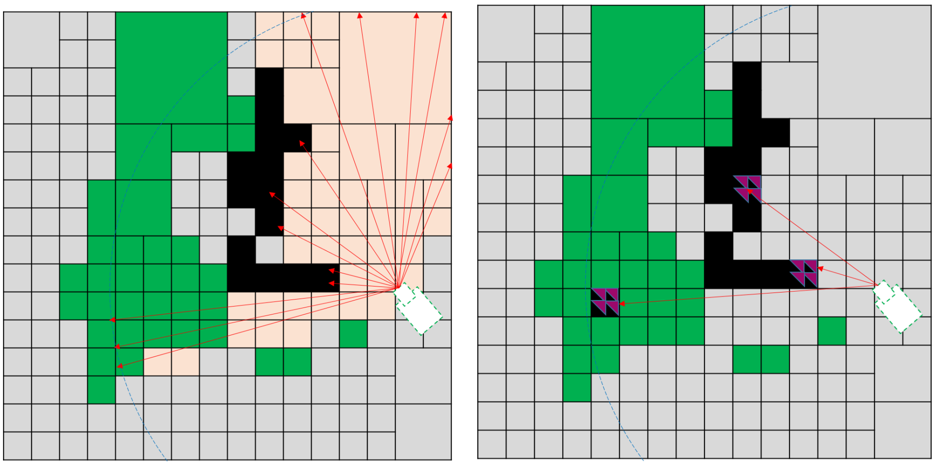
$$I_c(x) = \begin{cases} 1 & c(x) < c_{threshold} \\ 0 & \text{otherwise} \end{cases}, \tag{4}$$

  where $c(x)$ is the semantic confidence value fused by the deep learning model, and $c_{threshold}$ is a predefined confidence threshold.

We propose semantic information $I_{svv}(x)$ for a single voxel as a voxel that is visible, semantically labeled, and its confidence value less than a predefined threshold. The $I_{svv}(x)$ is defined as:

$$I_{svv}(x) = I_c(x) * I_{vis}(x) \tag{5}$$

Finally, the gain obtained from the proposed utility function is calculated by substituting $I_{svv}(x)$ into $I_{sem}(x)$ in Equation (1), where $\alpha = 0.3$ and $\beta = 0.7$. The total gain for a single view combines the volumetric and semantic info for all the voxels in the view. Figure 10 illustrates the SVV function which is a combination(accumulation) of both the volumetric in formation in Figure 10a and semantic information in Figure 10b.



(**a**) Volumetric Gain Illustration (**b**) Semantic Voxels Information $I_{svv}(x)$

Figure 10. SVV Utility function illustration: occupied (black), free (green), unknown (gray), and rays (red lines), sensor range (dashed blue), visible unknown (pink cubes).

- **Semantic Visited Object Of Interest Visible Voxel (SVOI)**

The proposed SVOI multi-objective function tends to direct the robot toward the views that contain occupied voxels classified as an object of interest but not visited sufficiently. In this function, the number of visits for each voxel is recorded to be used in the semantic information calculation. Let $I_s(x)$ be the indicator if the object belongs to the set of interest object S as follows:

$$I_s(x) = \begin{cases} 1 & x \in S_{obj} \\ 0 & otherwise \end{cases} \tag{6}$$

Each voxel in the semantic map holds a value $x_{numOfVisits}$ that represents the number of times it has been visited. In addition, let I_{obj} be the indicator of visiting threshold:

$$I_{obj}(x) = \begin{cases} 1 & x_{numOfVisits} < x_{threshold} \\ 0 & otherwise \end{cases} \tag{7}$$

where $x_{threshold}$ is a predefined threshold for the sufficient number of visiting.

We propose semantic information $I_{svio}(x)$ for a single voxel as a voxel that visible voxels, semantically labeled, belongs to object of interest, and their visiting value is less than a predefined threshold. It is defined as:

$$I_{svio}(x) = I_s(x) * I_{obj}(x) * I_{vis}(x) \tag{8}$$

Finally, the gain obtained from the proposed utility function is calculated by substituting $I_{svio}(x)$ into $I_{sem}(x)$ in Equation (1) where $\alpha = 0.3$ and $\beta = 0.7$. The total gain for a single view combines the volumetric and semantic info for all the voxels in the view.

3.3.3. Termination

The exploration task finishes when a termination condition is satisfied. Most of the termination condition depends on the scenario, such as victims' detection [51], where the number of iterations surpasses a predefined number of iterations [52], and information gain for all sampled viewpoints falls below a particular threshold of iterations [45]. Additionally, for frontier-based methods, the exploration terminates when no more frontiers are found [46].

The aim of the exploration is to label all the objects in the environment, and the termination condition proposed is when all the voxels are identified with a confidence value greater than the confidence threshold.

4. Experimental Setup and Results

4.1. Experimental Setup

4.1.1. Scenario

The main goal of the proposed approach is to explore the environment and label all the detected object autonomously. For each scenario, the robot begins in a starting location, and the entire process runs until a termination condition is satisfied. In the proposed scenario, takeoff and rotation steps are performed initially to create an initial map to allow the planner to generate sample viewpoints for evaluation.

The viewpoint sampling is performed using an incremental approach, specifically, rapidly-exploring random tree (RRT), where a series of points are selected randomly in a tree-like manner, instead of multiple single points for evaluation. Then, the viewpoints in each branch are evaluated using different utility functions for each test separately. After that, the branch that maximizes the gain is selected; the first edge is executed. The robot then navigates to the selected point and the process terminates when the number of exploration iteration exceeds a predefined number of iterations.

4.1.2. Simulation

Simulation experiments were performed on an ASUS laptop (Intel Core i7 @ 2.8 GHz x 8, 16 GB RAM). The NBV framework was implemented on Ubuntu 16.04 using the Robot Operating System (ROS kinetic) to handle message sharing and ease the transition to hardware. The gazebo was used to perform the actual simulation, with programming done in both C++ and Python. The simulations were performed using a UAV equipped with one RGB-D camera. The specifications of the camera are shown in Table 1. A collision box was constructed around the UAV to check for collisions with the environment and constrained within a work-space of size 0.25 m, 0.25 m, 0.25 m.

Table 1. Camera parameters.

Specification	Value
Horizontal FOV	60°
Vertical FOV	45°
Resolution	0.1 m
Range	5 m
Mounting orientation (Roll, Pitch, Yaw)	(0, 0, 0)

The gazebo environment, shown in Figure 11, was used as the unknown environment that the robot should explore. The simulation environment has the dimensions of $9.2m, 8.2m, 2.5m$ of multi-connected rooms with a corridor with several objects placed inside the rooms. The environment contains 11 objects, which are walls, floors, three people, a sofa, two chairs, a book shelve, a vase, and a table. The constructed maps are based on 3D occupancy grid using OctoMap library [41] with $res = 0.15m$ per pixel.

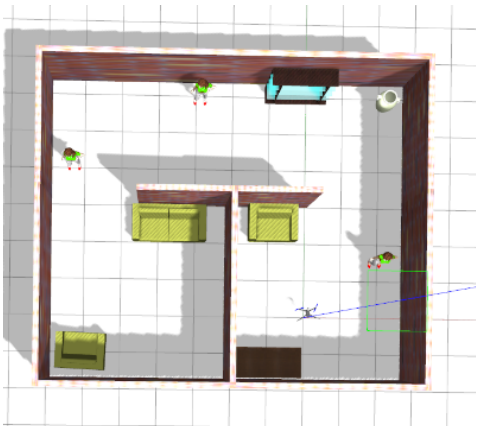


Figure 11. Simulation environment.

Each utility function was tested separately under controlled environments, and the following assumptions were made:

- The position of the robot was considered entirely known from the gazebo.
- The explored environment is unknown with known boundaries only.

4.1.3. Baselines for Comparison

For comparison baselines, we use the following utility functions for our evaluation: volumetric utility [42], pure entropy [49], average entropy [49], occlusion aware [49], and unobserved voxels [49]. The different utility functions are shown in Table 2.

Table 2. Utility functions.

Utility Function	$I_{vol}(x)$	$I_{sem}(x)$
Volumetric gain	[42]	0
Pure Entropy	[49]	0
Average Entropy	[50]	0
Occlusion Aware	[49]	0
Unobserved Voxels	[49]	0
Proposed SVV	[42]	Equation (5)
Proposed SVOI	[42]	Equation (8)

4.1.4. Evaluation Metrics

The evaluation metrics used in this work are the volumetric coverage, the number of detected objects, and the number of sufficiently labeled objects. The sufficiently labeled means that either the voxels have been visited multiple times until their confidence values increase above a certain threshold which is assigned to 0.7 or the number of visits exceeds a predefined threshold, which is assigned to three visits. This results in labeling the voxels correctly. Table 3 summarizes the evaluation metrics used.

Table 3. Evaluation metrics.

Coverage	Percentage of the number of known voxels compared with the total number of voxels the environment can cover. After each iteration, the coverage is calculated as follows $VC = \frac{Free+Occupied}{Free+Occupied+Unknown}$
Detected objects	Counting the number of detected objects in the environment
Sufficiently labeled objects	Counting the number of objects that are sufficiently labeled using the semantic color table
Total voxel type	Count the voxel for each category - Free - Occupied Low confidence - Occupied High confidence - Unknown

4.2. Experiment Results and Analysis

4.2.1. Mapping Evaluation

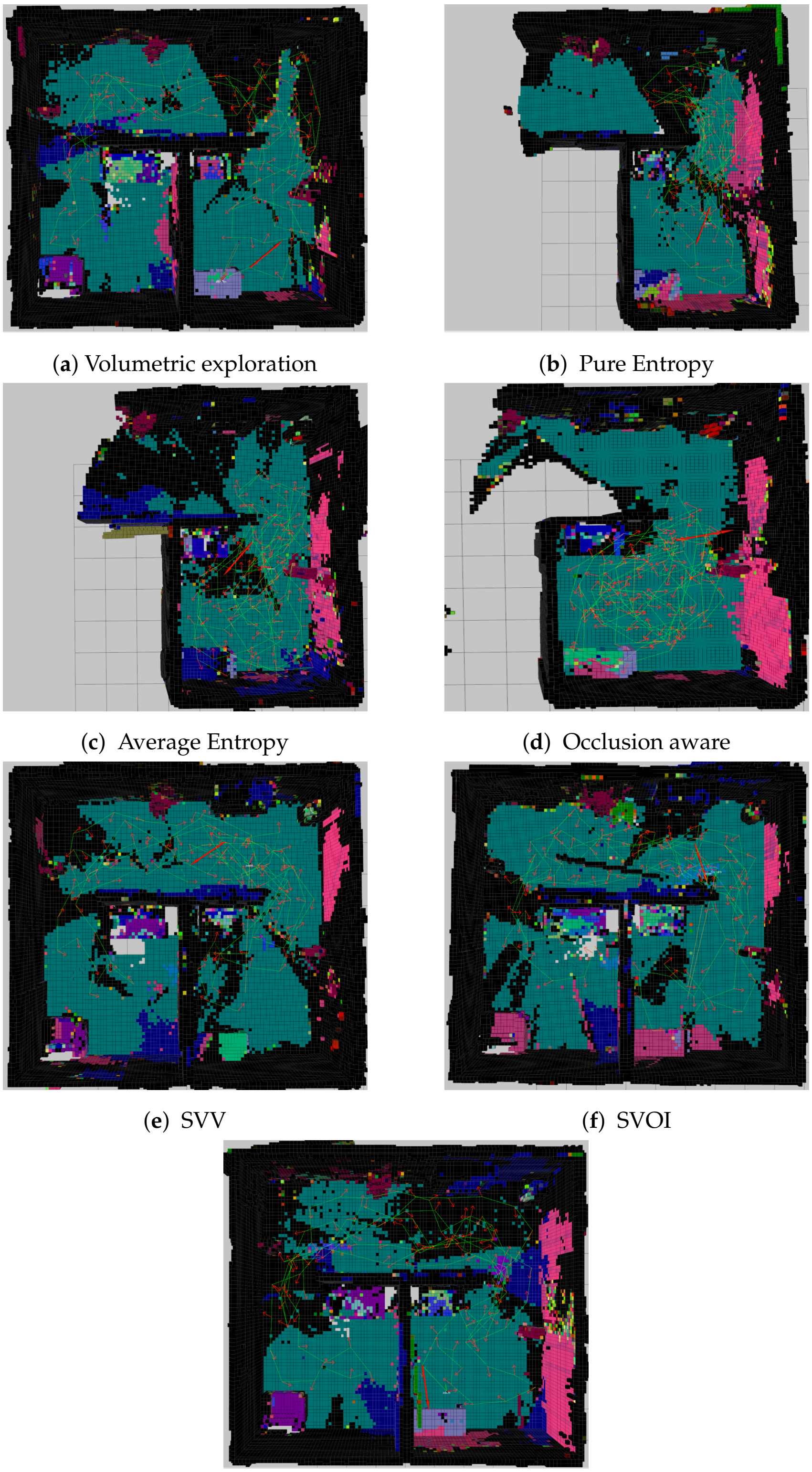
The results for seven different constructed 3D semantically-aware occupancy maps using the different utility functions are presented in Figure 12. The presented maps are obtained after 120 iterations. From Figure 12, the volumetric Gain, SVV, and SVOI utility functions were capable of constructing most of the map and covered almost above 85% of the map. However, the pure entropy, average entropy, and occlusion aware failed to construct the map within the predefined number of iterations.

4.2.2. Area Coverage Over Number of Iteration

The two proposed utility functions are compared with the state of the art commonly used utility functions. In both of the proposed utility functions, α was set to 0.3 and β was set to 0.7 to steer the utility towards semantic labeling rather than volumetric exploration. The reported results are for seven different experiments simulation tests. The tests are divided according to the viewpoint evaluation approaches. The exploration terminates after 120 iterations, and the semantic fusion method used in this test is the max fusion. Table 4 summarizes the results obtained. Figure 13 shows the volumetric coverage plot vs. the number of iteration for all the tests. The SVV utility function almost reached the same volumetric coverage as the SVOI and the volumetric utility functions. The SVOI and volumetric utility functions reached higher volumetric coverage in less number of iterations.

Table 4. Evaluation results.

NUM	**View Point Generation**	**Utility**	**Volumetric Coverage (%)**	**Num of Detected Objects**	**Num of Sufficiently Visited**
1	Receding Horizon RRT	Volumetric Gain	91.396	11	8
2		Pure Entropy	54.3802	8	5
3		Average Entropy	49.7883	8	5
4		Occlusion Aware	44.8293	8	5
5		Unobserved Voxel	75.8496	10	6
6		SVV	88.5718	11	7
7		SVOI	**93.1325**	11	8
8		SVOI (person detection)	92.0332	11	**9**



(**a**) Volumetric exploration

(**b**) Pure Entropy

(**c**) Average Entropy

(**d**) Occlusion aware

(**e**) SVV

(**f**) SVOI

(**g**) SVOI (person)

Figure 12. Generated semantic maps using different utility functions after 120 iterations. The colors correspond to semantic classes, the green line segments represent the path travelled, and the red arrows represent the viewpoints along the path.

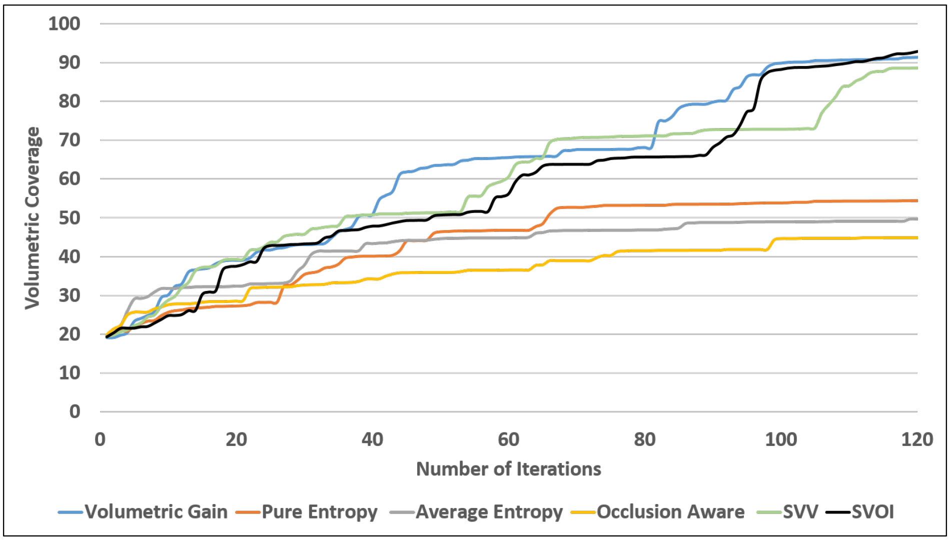


Figure 13. Volumetric coverage reached by different utilities using the same number of iterations.

4.2.3. Object Detection and Labeling

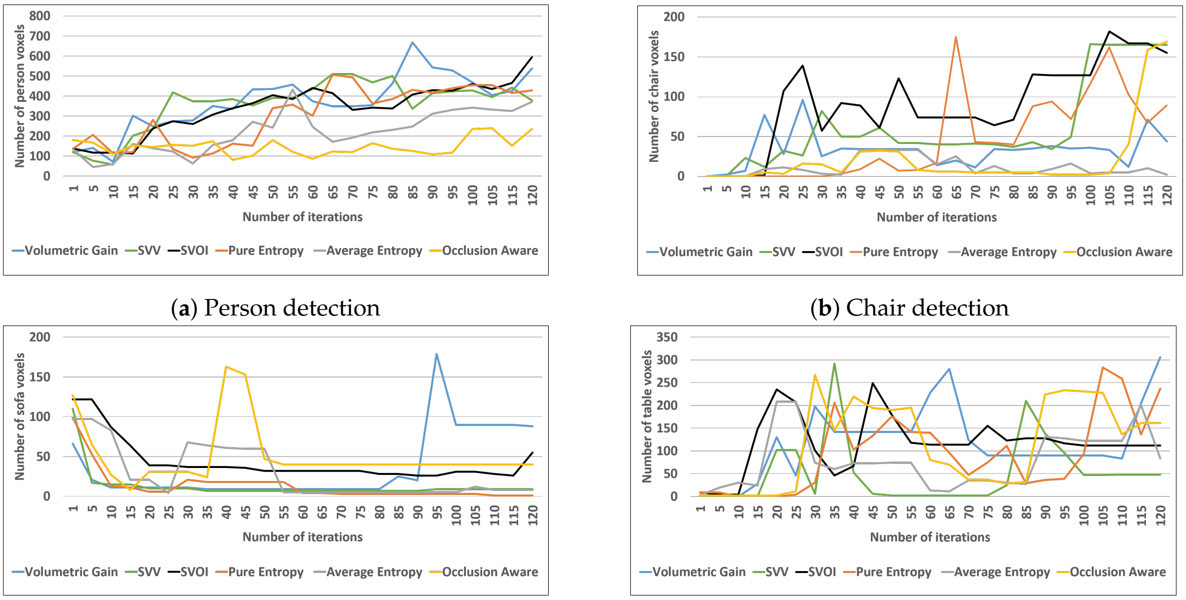
The color map shown previously in Figure 4 is used to show the detected objects. The volumetric gain, SVV, and SVOI utility function were able to detect all the objects in the environment. They were able to label more than 75% of objects sufficiently. Figure 14 shows four different object detections at the voxel level. The proposed SVOI function was capable of detecting a larger number of person voxels in less number of iteration, as shown in Figure 14a. In addition, the SVOI function showed a good performance when detecting chair voxels compared to other utility functions, as shown in Figure 14b.

The number of voxels for each object is recorded at each iteration until the exploration process terminates (120 iterations). Based on the observations from Figure 14, the recorded number of voxels for each object varies. The variation is due to the semantic fusion method, max fusion, which selects the semantic label with the highest confidence for the voxel. In addition, max fusion switches the labels of the voxel if the confidence value is marginal at the end of an iteration.

4.3. Object Search Application

The SVOI utility is an adaptive function where we can select the object of interest. In this test, rather than choosing the entire dataset to form interest objects, only the person selected to be the object of interest. Hence, in search and rescue context, this test aims to search for victims, locate them, and project their locations in the 3D semantic map. The results compared to the state of the art utility functions in terms of volumetric coverage, constructed semantic map and the ability to detect persons in less number of iterations. Any object from the dataset can be chosen as an object of interest.

Figure 15 shows that the proposed utility SVOI function outperforms the benchmark utility functions by covering 85 % of the environment in less number of iterations. Figure 12g shows the reconstructed map using the proposed function using the person as object of interest. The results show that the proposed utility function was able to sufficiently label one additional object (person) as shown in Table 4. Finally, Figure 16 shows that the proposed utility function performed better in detecting a person (ex. victim) compared with the other utilities in this study.



(**a**) Person detection

(**b**) Chair detection

(**c**) Sofa detection

(**d**) Table detection

Figure 14. Number of voxels labelled for each object using different utility functions.

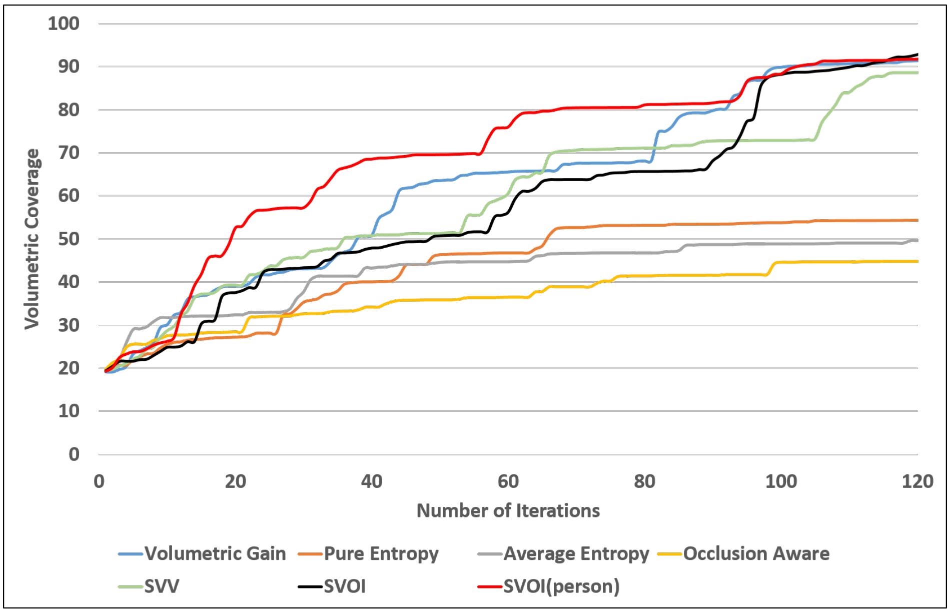


Figure 15. Volumetric coverage (proposed utility SVOI where the object of interest is the person).

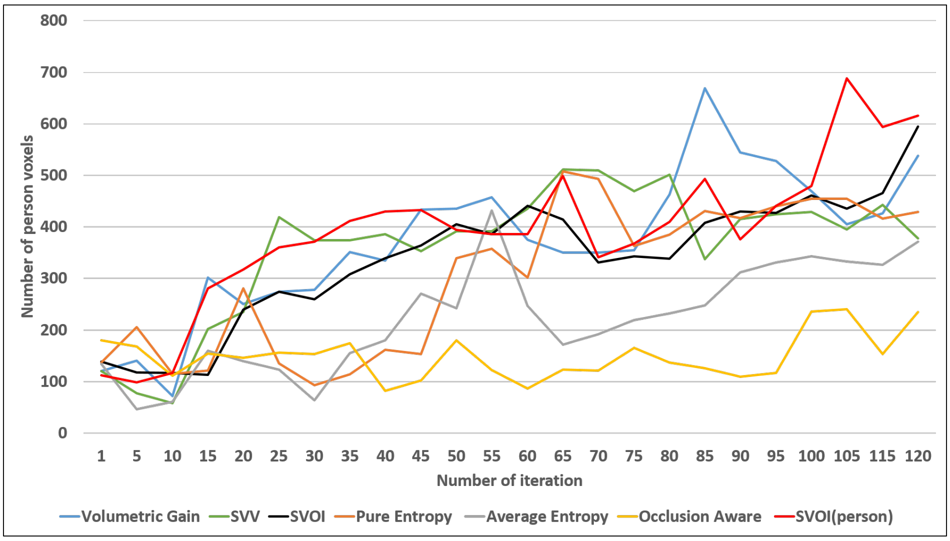


Figure 16. Person detection at the voxel level.

5. Conclusions

In this paper, a strategy for autonomous, semantically-aware exploration for object labeling and 3D semantic mapping was presented. This strategy facilitates exploring and mapping unknown environments efficiently, while directing the robot to label the objects present in the environment. In this work, we proposed a novel 3D, semantically-aware occupancy map data structure that carries the occupancy information as well as the semantic specific annotations to be utilized in the autonomous exploration task. The mapping method employs a deep learning model to segment objects semantically, generate annotated 3D point cloud, and create a 3D, occupied, semantically-aware map. Two new multi-objective utility functions were introduced to quantify the contextual information from the 3D semantic-occupied map to explore unknown environments.

Experimental results in the simulation demonstrated that the semantically-aware exploration and mapping was successfully able to explore the environment and label the objects by utilizing the contextual information from the 3D semantic map. The proposed utility functions showed a reliable performance in exploration and object labeling. Comparison between the proposed utility functions and the state of the art utility functions is provided: the proposed utility outperforms the benchmark utilities by volumetric coverage and accurate labeling.

The semantic fusion method used in the proposed strategy suffers from inaccuracies (frequent label shifts) when the confidence in the semantic labeling is on the border between multiple classes. This will be addressed in future work by improving the accuracy of the semantic labeling and adapting a variation of the Bayesian probabilistic semantic fusion. Additionally, the utility function introduced requires careful tuning of the scaling variables α and β; we will explore in the future, the possibility of introducing an adaptive scaling strategy that accommodates for various objectives and environments.

Author Contributions: Conceptualization, R.A., T.T., and N.A.; methodology, R.A. and T.T.; software, R.A. and T.T.; validation, R.A.; formal analysis, R.A.; investigation, R.A.; resources, R.A.; data curation, R.A.; writing—original draft preparation, R.A.; writing—review and editing, R.A.; visualization, R.A.; supervision, T.T., J.M.M.D., L.S., and N.A.; project administration, T.T., J.M.M.D., L.S., and N.A.; funding acquisition. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by ICT fund for "A Semi Autonomous Mobile Robotic C4ISR System for Urban Search and Rescue" project.

Acknowledgments: This publication acknowledges the support by the Khalifa University of Science and Technology under Award Number RC1-2018-KUCARS.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Naidoo, Y.; Stopforth, R.; Bright, G. Development of an UAV for search & rescue applications. In Proceedings of the IEEE Africon'11, Livingstone, Zambia, 13–15 September 2011; pp. 1–6.
2. Erdelj, M.; Natalizio, E.; Chowdhury, K.R.; Akyildiz, I.F. Help from the sky: Leveraging UAVs for disaster management. *IEEE Pervasive Comput.* **2017**, *16*, 24–32. [CrossRef]
3. Waharte, S.; Trigoni, N. Supporting search and rescue operations with UAVs. In Proceedings of the 2010 International Conference on Emerging Security Technologies, Canterbury, UK, 6–7 September 2010; pp. 142–147.
4. Hallermann, N.; Morgenthal, G. Visual inspection strategies for large bridges using Unmanned Aerial Vehicles (UAV). In Proceedings of the 7th IABMAS, International Conference on Bridge Maintenance, Safety and Management, Shangai, China, 7–11 July 2014; pp. 661–667.
5. Wada, A.; Yamashita, T.; Maruyama, M.; Arai, T.; Adachi, H.; Tsuji, H. A surveillance system using small unmanned aerial vehicle (UAV) related technologies. *NEC Tech. J.* **2015**, *8*, 68–72.
6. Lang, D.; Paulus, D. Semantic Maps for Robotics. In Proceedings of the Workshop" Workshop on AI Robotics" at ICRA, Chicago, IL, USA, 14–18 September 2014.
7. Cadena, C.; Carlone, L.; Carrillo, H.; Latif, Y.; Scaramuzza, D.; Neira, J.; Reid, I.; Leonard, J.J. Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. *IEEE Trans. Robot.* **2016**, *32*, 1309–1332. [CrossRef]
8. Kostavelis, I.; Gasteratos, A. Semantic mapping for mobile robotics tasks: A survey. *Robot. Auton. Syst.* **2015**, *66*, 86–103. [CrossRef]
9. Wurm, K.M.; Hornung, A.; Bennewitz, M.; Stachniss, C.; Burgard, W. OctoMap: A probabilistic, flexible, and compact 3D map representation for robotic systems. In Proceedings of the ICRA 2010 workshop on Best Practice in 3D Perception and Modeling for Mobile Manipulation, Anchorage, AS, USA, 3–7 May 2010; Volume 2.
10. Lai, K.; Bo, L.; Fox, D. Unsupervised feature learning for 3d scene labeling. In Proceedings of the 2014 IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China, 31 May–7 June 2014; pp. 3050–3057.
11. Pillai, S.; Leonard, J. Monocular slam supported object recognition. *arXiv* **2015**, arXiv:1506.01732.
12. Salas-Moreno, R.F.; Newcombe, R.A.; Strasdat, H.; Kelly, P.H.; Davison, A.J. Slam++: Simultaneous localisation and mapping at the level of objects. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 1352–1359.
13. Vineet, V.; Miksik, O.; Lidegaard, M.; Nießner, M.; Golodetz, S.; Prisacariu, V.A.; Kähler, O.; Murray, D.W.; Izadi, S.; Pérez, P.; et al. Incremental dense semantic stereo fusion for large-scale semantic scene reconstruction. In Proceedings of the 2015 IEEE International Conference on Robotics and Automation (ICRA), Seattle, WA, USA, 26–30 May 2015; pp. 75–82.
14. Kostavelis, I.; Charalampous, K.; Gasteratos, A.; Tsotsos, J.K. Robot navigation via spatial and temporal coherent semantic maps. *Eng. Appl. Artif. Intell.* **2016**, *48*, 173–187. [CrossRef]
15. Yamauchi, B. A frontier-based approach for autonomous exploration. In Proceedings of the Proceedings 1997 IEEE International Symposium on Computational Intelligence in Robotics and Automation CIRA'97. 'Towards New Computational Principles for Robotics and Automation', Monterey, CA, USA, 10–11 July 1997; pp. 146–151.
16. Elfes, A. Using occupancy grids for mobile robot perception and navigation. *Computer* **1989**, *22*, 46–57. [CrossRef]
17. Connolly, C. The determination of next best views. In Proceedings of the 1985 IEEE International Conference on Robotics and Automation, St. Louis, MO, USA, 25–28 March 1985; Volume 2, pp. 432–435.
18. Półka, M.; Ptak, S.; Kuziora, Ł. The use of UAV's for search and rescue operations. *Procedia Eng.* **2017**, *192*, 748–752. [CrossRef]
19. Tang, J.; Zhu, K.; Guo, H.; Liao, C.; Zhang, S. Simulation optimization of search and rescue in disaster relief based on distributed auction mechanism. *Algorithms* **2017**, *10*, 125. [CrossRef]
20. Goian, A.; Ashour, R.; Ahmad, U.; Taha, T.; Almoosa, N.; Seneviratne, L. Victim Localization in USAR Scenario Exploiting Multi-Layer Mapping Structure. *Remote Sens.* **2019**, *11*, 2704. [CrossRef]
21. Lindeberg, T. Scale invariant feature transform. *Scholarpedia* **2012**, *7*, 10491. [CrossRef]

22. Dalal, N.; Triggs, B. Histograms of oriented gradients for human detection. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 20–25 June 2005; Volume 1, pp. 886–893.
23. Wang, L. *Support Vector Machines: Theory and Applications*; Springer Science & Business Media: Berlin, Germany, 2005; Volume 177.
24. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. Ssd: Single shot multibox detector. In Proceedings of the European conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; Springer: Berlin, Germany, 2016; pp. 21–37.
25. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Amsterdam, The Netherlands, 8–16 October 2016; pp. 779–788.
26. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
27. Redmon, J.; Farhadi, A. YOLO9000: better, faster, stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7263–7271.
28. Zhao, H.; Shi, J.; Qi, X.; Wang, X.; Jia, J. Pyramid scene parsing network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2881–2890.
29. Qi, C.R.; Su, H.; Mo, K.; Guibas, L.J. Pointnet: Deep learning on point sets for 3d classification and segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 652–660.
30. Qi, C.R.; Yi, L.; Su, H.; Guibas, L.J. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *Advances in Neural Information Processing Systems*; The MIT Press: Cambridge, MA, USA, 2017; pp. 5099–5108.
31. Qi, C.R.; Liu, W.; Wu, C.; Su, H.; Guibas, L.J. Frustum pointnets for 3d object detection from rgb-d data. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 918–927.
32. Rosinol, A.; Abate, M.; Chang, Y.; Carlone, L. Kimera: An Open-Source Library for Real-Time Metric-Semantic Localization and Mapping. *arXiv* **2019**, arXiv:1910.02490.
33. Dang, T.; Papachristos, C.; Alexis, K. Visual Saliency-Aware Receding Horizon Autonomous Exploration with Application to Aerial Robotics. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, QLD, Australia, 21–25 May 2018; pp. 2526–2533. [CrossRef]
34. Dang, T.; Papachristos, C.; Alexis, K. Autonomous exploration and simultaneous object search using aerial robots. In Proceedings of the 2018 IEEE Aerospace Conference, Big Sky, MT, USA, 3–10 March 2018.
35. Heng, L.; Gotovos, A.; Krause, A.; Pollefeys, M. Efficient visual exploration and coverage with a micro aerial vehicle in unknown environments. In Proceedings of the 2015 IEEE International Conference on Robotics and Automation (ICRA), Seattle, WA, USA, 26–30 May 2015; pp. 1071–1078.
36. Fraundorfer, F.; Heng, L.; Honegger, D.; Lee, G.H.; Meier, L.; Tanskanen, P.; Pollefeys, M. Vision-based autonomous mapping and exploration using a quadrotor MAV. In Proceedings of the 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, Vilamoura, Portugal, 7–12 October 2012; pp. 4557–4564.
37. Cieslewski, T.; Kaufmann, E.; Scaramuzza, D. Rapid Exploration with Multi-Rotors: A Frontier Selection Method for High Speed Flight. In Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, 24–28 September 2017.
38. Song, S.; Lichtenberg, S.P.; Xiao, J. Sun rgb-d: A rgb-d scene understanding benchmark suite. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Diego, CA, USA, 20–25 June 2005; pp. 567–576.
39. Zhou, B.; Zhao, H.; Puig, X.; Fidler, S.; Barriuso, A.; Torralba, A. Scene parsing through ade20k dataset. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 633–641.
40. Xuan, Z.; David, F. Real-Time Voxel Based 3D Semantic Mapping with a Hand Held RGB-D Camera. 2018. Available online: https://github.com/floatlazer/semantic_slam (accessed on 3 March 2020).
41. Hornung, A.; Wurm, K.M.; Bennewitz, M.; Stachniss, C.; Burgard, W. OctoMap: An efficient probabilistic 3D mapping framework based on octrees. *Auton. Robots* **2013**, *34*, 189–206. [CrossRef]

42. Bircher, A.; Kamel, M.; Alexis, K.; Oleynikova, H.; Siegwart, R. Receding Horizon "Next-Best-View" Planner for 3D Exploration. In Proceedings of the 2016 IEEE International Conference on Robotics and Automation (ICRA), Stockholm, Sweden, 16–21 May 2016; pp. 1462–1468.
43. Burgard, W.; Moors, M.; Stachniss, C.; Schneider, F.E. Coordinated multi-robot exploration. *IEEE Trans. Robot.* **2005**, *21*, 376–386. [CrossRef]
44. Stachniss, C.; Grisetti, G.; Burgard, W. Information Gain-based Exploration Using Rao-Blackwellized Particle Filters. *Robot. Sci. Syst.* **2005**, *2*, 65–72.
45. Paul, G.; Webb, S.; Liu, D.; Dissanayake, G. Autonomous robot manipulator-based exploration and mapping system for bridge maintenance. *Robot. Auton. Syst.* **2011**, *59*, 543–554. [CrossRef]
46. Al khawaldah, M.; Nuchter, A. Enhanced frontier-based exploration for indoor environment with multiple robots. *Adv. Robot.* **2015**, *29*. [CrossRef]
47. Karaman, S.; Frazzoli, E. Sampling-based algorithms for optimal motion planning. *Int. J. Robot. Res.* **2011**, *30*, 846–894. [CrossRef]
48. Lavalle, S.M. *Rapidly-Exploring Random Trees: A New Tool for Path Planning*; Technical Report; Iowa State University: Ames, IA, USA, 1998.
49. Delmerico, J.A.; Isler, S.; Sabzevari, R.; Scaramuzza, D. A comparison of volumetric information gain metrics for active 3D object reconstruction. *Auton. Robot.* **2018**, *42*, 197–208. [CrossRef]
50. Kriegel, S.; Rink, C.; Bodenmüller, T.; Suppa, M. Efficient next-best-scan planning for autonomous 3D surface reconstruction of unknown objects. *J. Real-Time Image Process.* **2015**, *10*, 611–631. [CrossRef]
51. Batista, N.C.; Pereira, G.A.S. A Probabilistic Approach for Fusing People Detectors. *J. Control Autom. Electr. Syst.* **2015**, *26*, 616–629. [CrossRef]
52. Isler, S.; Sabzevari, R.; Delmerico, J.; Scaramuzza, D. An information gain formulation for active volumetric 3D reconstruction. In Proceedings of the 2016 IEEE International Conference on Robotics and Automation (ICRA), Stockholm, Sweden, 16–21 May 2016; pp. 3477–3484. [CrossRef]