robotics

MDPI

Article

Probabilistic Allocation of Specialized Robots on Targets Detected Using Deep Learning Networks

Omar Al-Buraiki, Wenbo Wu and Pierre Payeur *

School of Electrical Engineering and Computer Science, University of Ottawa, 800 King Edward, Ottawa, ON K1N 6N5, Canada; oalbu060@uottawa.ca (O.A.-B.); wwu039@uottawa.ca (W.W.)

* Correspondence: ppayeur@uottawa.ca; Tel.: +1-613-562-5800

Received: 31 May 2020; Accepted: 14 July 2020; Published: 16 July 2020

Abstract: Task allocation for specialized unmanned robotic agents is addressed in this paper. Based on the assumptions that each individual robotic agent possesses specialized capabilities and that targets representing the tasks to be performed in the surrounding environment impose specific requirements, the proposed approach computes task-agent fitting probabilities to efficiently match the available robotic agents with the detected targets. The framework is supported by a deep learning method with an object instance segmentation capability, Mask R-CNN, that is adapted to provide target object recognition and localization estimates from vision sensors mounted on the robotic agents. Experimental validation, for indoor search-and-rescue (SAR) scenarios, is conducted and results demonstrate the reliability and efficiency of the proposed approach.

Keywords: task allocation; multi-agent systems; specialized robots; probabilistic representation; target object detection; deep learning; Mask R-CNN; classification; segmentation

1. Introduction

This paper introduces a mechanism that explores the concept of specializing individual robotic agents to respond to constrained tasks. A formalism is designed for task allocation in the context of a collaborative multi-robot swarm. Unlike previous works that consider heterogeneity among the robotic agents mainly from their physical construction, here a specific definition of the individual robots' specialization is formulated. It leverages the embedded hardware and software characteristics of each agent and the estimation of requirements imposed by specific target objects. As a result, an advanced form of specialized labor division emerges in the swarm, which distributes the labor among the individual agents based on best matching the tasks' specific requirements to each robot's specialized capabilities. This form of task allocation can increase the net efficiency of the robotic swarm. In this paper, a probabilistic approach is proposed to compute the fit of the individual agents amongst the robotic swarm, based on matching their specialized capabilities with the corresponding requirements imposed by the tasks. The latter take the form of visually recognized target objects in the environment surrounding the robots.

For such a task allocation mechanism to be robust, recent developments in the field of artificial intelligence are leveraged and a deep learning method named Mask R-CNN is adopted to recognize and segment target objects in unstructured environments from vision sensors mounted on autonomous robots. Reliable target object detection supports efficient and responsive automated task allocation for specialized unmanned robotic systems.

The proposed approach addresses the problem of task allocation in swarm robotics in the specific context where specialized capabilities of the individual agents are considered. It is based on the assumption that each individual agent possesses specialized functional capabilities and that the expected tasks, which are distributed in the surrounding environment, impose specific requirements.

A task allocation mechanism is formulated to compute the specialty-based task allocation probabilities of the individual agents, with the purpose to assign the qualified agents to the corresponding detected tasks. The selection of an agent is based on the probabilistic matching between the individual agents' specialized capabilities and the constraints (i.e., requirements) that are imposed by the detected targets.

The formulation of the proposed approach evolves through four stages of development. First, a deep learning method using Mask R-CNN architecture serves to recognize target objects in unstructured environments from vision sensors mounted on autonomous robots. It is implemented to represent a robust target objects recognition stage. The output of the sensing layer drives the proposed task allocation scheme. Second, a matching scheme is developed, to best match each agent's specialized capabilities with the corresponding detected tasks. At this stage, a binary definition of agents' specialization serves as the basis for task-agent association. Third, the task-agent matching scheme is expanded to an innovative probabilistic specialty-based task-agent allocation framework to exploit the potential of agents' specialization consideration in a standardized format. Fourth, a coordination scheme is implemented to coordinate the qualified individuals to respond to the detected tasks. In this stage of development, the agents' availability state is considered along with their specialty, to improve the proposed system's reliability to accomplish the mission goals even when the most specialized agents that possess a high level of competences are not available or busy with another task. In such case, the system is designed to show robustness and automatically substitute the most qualified agents with other specialized agents that are available, even though the latter may offer a lower level of competence. The proposed approach can allocate the specialized qualified agents to the corresponding tasks with versatility, based on the requirements of the application, either with only the most specialized agent considered or with all qualified agents when the intervention of a group of agents is desirable.

2. Related Work

Previous literature extensively addressed multi-agent task allocation to map robotic agents to corresponding tasks [1,2]. Jones and Mataric [3] proposed a task-agent assignment approach and built a state transition probabilistic model to respond to changing tasks. A task allocation probabilistic grid assignment algorithm was introduced in [4]. The approach partitions the targets environment to a grid of cells, then assigns the available robots in each cell to allocate the targets that occupy the same cell. Claes et al. [5] used a Markov decision process to address the task-agent assignment as a spatial task planning problem. Yasuda et al. [6] introduced a probabilistic model based on a response threshold to control the individual agents to perform food foraging processes. The proposed model allows for the robots that have probabilities exceeding a specific threshold to leave the nest and search for food. Recently, Wu et al. [7] proposed a task allocation probabilistic model based on environmental stimulus and the agent's response threshold. A general architecture of a task allocation approach for multi-agent systems under uncertainty is also investigated through an empirical study in [8]. Four task-allocation strategies are empirically compared to investigate the task allocation handling problem. The results show that the task allocation is changing and the system's overall performance is a function of noise.

On the other hand, environment monitoring systems have been combined with multi-agent systems [9] to support realistic application of robots' interaction with their surrounding environment. Feature extraction and object class recognition on target objects that robotic agents encounter, while exploring a workspace, play a critical role for a reliable estimation of the specialized agents' qualification to intervene on the detected targets. The application of convolutional neural networks (CNNs) to image recognition [10] and object localization [11] significantly improved the accuracy of object detection. Alternative deep learning methods solving target object detection problems were previously investigated as part of this research [12]. These include Faster R-CNN [13], which is a region-based convolutional neural network that provides a class-level detection, and Mask R-CNN [14], which detects specific instances of different classes of objects in an image and generates an image map that highlights the pixel distribution of each instance.

3. Proposed Framework

The central objective of this research is to leverage vision sensors embedded on unmanned robotic agents to estimate the characteristics of target objects found in the environment and toward which specific agents will be directed. The requirements imposed by a detected task to be performed, associated with the physical characteristics of a given target object, should drive the response of specific robotic agents possessing adequate physical construction characteristics or equipped with specific embedded devices. The concept of specialization of the robotic agents forms the central consideration around which the solution is designed, with the goal to systematically assign the most competent agent to intervene in a given situation defined by a detected task, while benefitting from the support of other robotic agents in a collaborative manner. Figure 1 provides a general overview of the proposed framework.

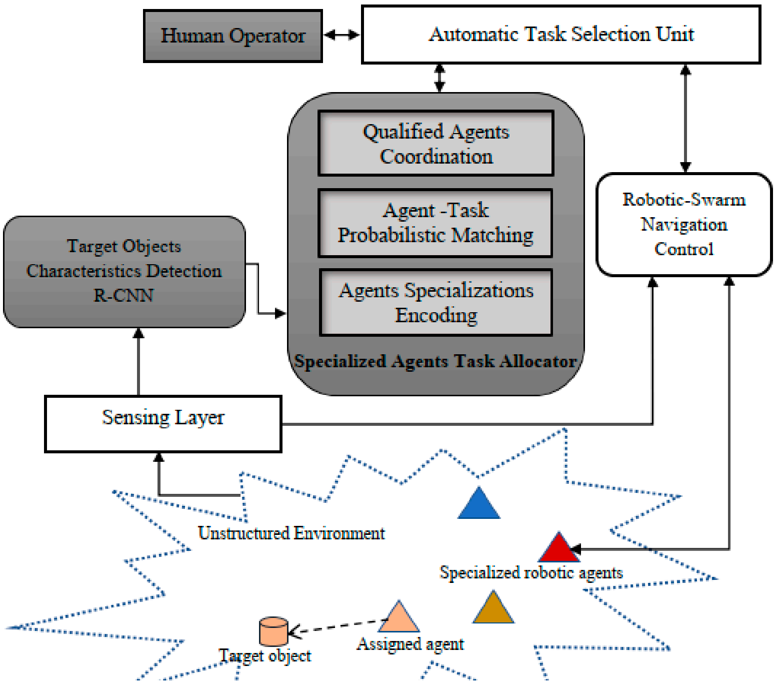

Human Operator
Automatic Task Selection Unit
Qualified Agents Coordination
Agent -Task Probabilistic Matching
Agents Specializations Encoding
Specialized Agents Task Allocator
Target Objects Characteristics Detection R-CNN
Robotic-Swarm Navigation Control
Sensing Layer
Unstructured Environment
Specialized robotic agents
Target object
Assigned agent

Figure 1. General framework for specialized task-agent allocation.

To achieve this objective, a probabilistic task allocation scheme to match the most qualified specialized agents with the detected tasks is proposed and integrated with an object detection convolutional neural network stage. The solution is experimentally investigated as a framework for multi-agent robotic systems autonomous operation. The developments that are introduced in this paper are presented in gray boxes in Figure 1. The low-level robotic swarm controller that tackles the robots' dynamics and navigation, and the swarm's formation control, were introduced in [15], while the automatic task selection unit (ATSU) was proposed in our previous work [16]. The latter is responsible for the decision-making process, while remaining under high-level human supervision for strategic guidance, as depicted in Figure 1.

This work expands on our previous design and efficiently merges the detection of target objects' characteristics provided by modern deep learning recognition methods with original concepts for the specialization of individual robotic agents that form the grounds of a robust probabilistic task allocation process for multi-agent robotic systems.

4. Target Object Recognition

Target object detection aims at determining whether or not instances of objects from predefined categories appear in an image collected by robotic agents and, if present, at estimating the spatial location and extent of each instance. The deep learning Mask R-CNN [14] architecture is selected as a target object detection module because of its class-level detection combined with pixel-precise mask segmentation capability that highlights the pixel distribution, and therefore the location, of each recognized class instance in an image. This characteristic is a key advantage compared with general target detectors. This provides significant benefits for autonomous robot navigation toward the target objects considered for task allocation. Mask R-CNN is a state-of-the-art two-stage detection framework. In the first stage, the region proposal network (RPN) [13] generates a set of regions of interests as potential bounding box candidates. Then, the second stage classifies the proposals, refines the bounding box and generates segmentation masks in parallel, where the mask prediction branch is a small fully convolutional network (FCN) [17]. Figure 2 illustrates the detailed two-stage structure of the Mask R-CNN architecture that was developed for our experiments on target object detection [12]. In this work, the target object detection module becomes an integral component of the specialized robotic agent task allocation process. Images are captured by vision sensors mounted on the robots and used as input to the target object detection module. The latter supports the detection of object characteristics with the CNN-deep learning network on every detected object. This network then serves as an input to the task allocator (Figure 1).

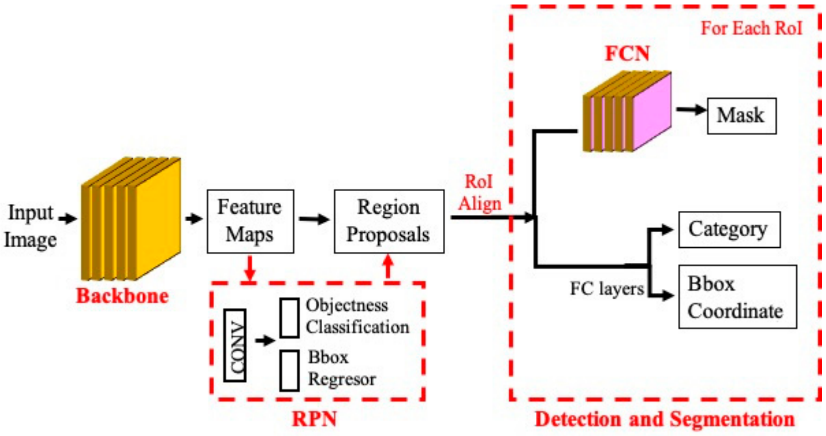


Figure 2. Detailed two-stage structure of Mask R-CNN architecture.

4.1. Deep Learning Network Training

Given that supervised learning is used to train and tune the CNN, only classes of objects that are included in the training are expected to be detected. As a result, the training of the CNN can be reconfigured and precisely adapted for various contexts of application. Targets objects considered in this study relate to indoor search-and-rescue (SAR) operations. The five classes considered include a person to be rescued, door to be opened, stairs to be climbed, posted signs or maps to be read to support robots navigation, and fire to be extinguished.

A corresponding dataset is developed for such SAR scenarios. It is composed of three parts. One part with 300 sample images is from the McIndoor20000 dataset [18] that contains sample images with pre-labelled categories of objects covering 3 different classes (doors, signs, and stairs). The second part with 195 additional sample images exemplifies persons and tv-monitors, which are here associated with the “fire” class for safety reasons. These images are extracted from the Pascal VOC 2007 dataset [19] that contains samples from 20 different classes. Sample images selected from that dataset are among the 632 items that also provide a bounding box and a segmentation mask annotation for the object instances. The third part is formed of 50 sample images, describing relatively complex situations, such as a door with a sign on it, and were captured by our team in real indoor environments. These additional samples are added to alleviate the inherent limitation associated with sample images from the McIndoor20000 dataset, that exhibit only a single instance of object in every image. All sample images are manually annotated with category label, bounding box and the corresponding segmentation mask information for each object instance through the LabelMe [20] annotation tool, except for images from the Pascal VOC 2007 dataset, since these are already segmented and labelled. The segmentation mask of each object instance is saved in the PNG image format. The bounding box coordinates are recorded in the JSON format file with the category label. The dataset formation process leads to a dataset size of 545 images, with a fair balance of samples representing each of the five classes considered. Table 1 details the number of samples in the training and validation datasets, for each of the five classes.

Table 1. Dataset samples distribution.

	Category	Number of Images	Training Set	Validation Set
Pre-labelled (195)	person	101	80	21
	tv-monitor (fire)	94	85	9
Manually labelled (350)	door	125	98	27
	sign	117	97	20
	stairs	108	100	8
Total	5 classes	545	460	85

For the implementation of the Mask R-CNN framework, the backbone architecture used for extracting features is ResNet-50 [21] and feature pyramid network (FPN) [22] with pre-trained weights on the Microsoft COCO dataset [23]. The head branches of the network are further adjusted and trained on the above dataset. Data augmentation, involving flipping, rotating, scaling, blurring, changing contrast, and lightness, is included to extend the variety of input samples, which enables to increase the generalization ability of the model. It helps to reduce the influence of input images' orientation and scale. The training is performed in three stages, as shown in Figure 3, that consist of: (i) fixing all layers except the head, and train the head part; (ii) unfreezing the layers in ResNet stage 4 and up, to train the region proposal part and head part; and (iii) unfreezing all layers and fine-tuning the whole model. During the whole process, a stochastic gradient descent (SGD) optimizer is used, with starting learning rate of 0.001, weight decay of 0.0001, momentum of 0.9, and gradient clip norm of 5.0.

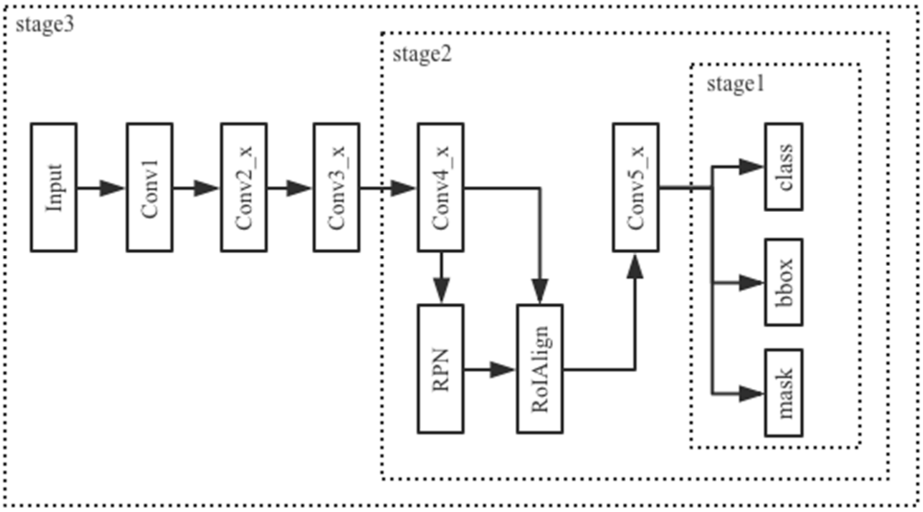


Figure 3. Three training stages.

All training processes are performed on an 8GB memory NVIDIA Tesla P4 GPU configured in virtual machine supported by Google Compute Engine. The trained weights of the detection model relevant with the SAR scenarios defined above is saved as .h5 file, which is easy to load offline. It enables the detection to be conducted separately from the GPU-based training network platform and run on an embedded CPU-based computer. This architecture makes it possible to integrate the detection and task allocation stages on the robotic platform and not remain dependent on a network connection.

4.2. Target Objects Detection

The inference results through the object detection module return the object class category and corresponding detection score for every detected object, which serves as an input to the proposed task allocator. The output information formed of the segmentation mask with bounding box on target objects supports robots' navigation and localization, which is introduced in our previous works [15,16], but it is beyond the scope of this paper. In general, the output of the object detection module is given by:

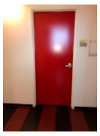
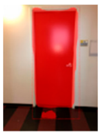
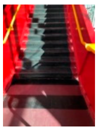
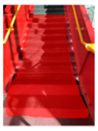
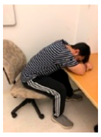
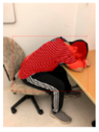
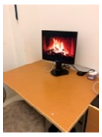
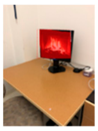
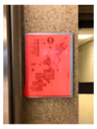
$$\hat{\boldsymbol{P}}_T = \begin{bmatrix} P_{C_1} & P_{C_2} & \cdots & P_{C_F} \end{bmatrix}^T \tag{1}$$

where $\hat{\boldsymbol{P}}_T$ represents an input to the proposed task allocator; F is the maximum number of features (or constraints) to be detected on the expected target objects. For the proposed SAR scenarios, five classes are considered: therefore, $F = 5$, leading to:

$$\hat{\boldsymbol{P}}_{T_{SAR}} = \left[P_{C_1}, P_{C_2}, P_{C_3}, P_{C_4}, P_{C_5}\right]^T \tag{2}$$

where C_k : k from 1 to F, respectively denote the classes of door, stairs, person, tv-monitor (fire), and signs respectively; $P_{C1} \sim P_{C5}$ are the recognition confidence scores on a target object associated with each class category. Table 2 shows examples of object detection estimates, along with the corresponding specialized functionalities expected of the robotic agents to tackle each class of target objects.

Table 2. Object detection and confidence on visual features (target object class) matched with related robots.

Input Image	Inference Results	Detected Object	Targets Detection Output	Agent Specialized Functionality
		door (C_1)	$\hat{P}_{T_{SAR}} = [0.995, 0, 0, 0, 0]$	Open doors
		stairs (C_2)	$\hat{P}_{T_{SAR}} = [0, 0.963, 0, 0, 0]$	Climb stairs
		person (C_3)	$\hat{P}_{T_{SAR}} = [0, 0, 0.958, 0, 0]$	Assist people
		tv-monitor (fire) (C_4)	$\hat{P}_{T_{SAR}} = [0, 0, 0, 0.954, 0]$	Extinguish fire
		sign (C_5)	$\hat{P}_{T_{SAR}} = [0, 0, 0, 0, 0.983]$	Read signs

5. Probabilistic Task Allocation Scheme

Specialized agents are expected to be allocated and respond to detected tasks when a given agent's specialty represents a sufficient fitting level to match with the detected task's requirements.

The latter correspond to the confidence level estimated by the target object detection stage. However, a given agent can qualify to be allocated to different tasks but with different fitting levels, encoded as probabilities. The proposed task allocation matching scheme leverages the output of the target objects detection defined in Equation (1) and performs two functions: 1) to compute the probabilistic specialty fitting level of the individual agents, introduced in Sections 5.1 and 5.2; and Section 2) to coordinate task allocation to match the detected tasks with the most qualified and available agents, as detailed in Sections 5.3 and 5.4.

5.1. Specialization Definition and Coding

A swarm of robots $\{R_i, i = 1,\ 2, \ldots, a\}$ consists of a, specialized individual agents, R_i, and provides F different specialized capabilities (i.e., in this case, the agents' specialized capabilities are considered equal to the number of constraints, or target object classes, F, that can be detected). The definition of an agent's specialization describes the presence or absence of specific hardware, or particular physical construction, that is essential to completing a given task (e.g., robotic hand to open a door, or stretcher for rescuing a person). The agent's specialty is encoded in an agent's specialty binary vector, $\boldsymbol{S}_i : \{\mathrm{s}_k, k = 1,\ 2, \ldots, F\}$, where, $\boldsymbol{S}_i \in \mathcal{R}^{1\times F}$. Each entry defined as $\mathrm{s}_k = 1$ means that the robot possesses the corresponding capability; $\mathrm{s}_k = 0$ indicates that the robot is not equipped with the corresponding capability to tackle a given requirement, X_k. Every requirement is meant to correspond to a given class, C_k, among F of them. Table 3 summarizes the characteristics of a group of seven robotic agents considered to experimentally evaluate the proposed approach in simulated SAR scenarios.

Table 3. Formulation of robotic agents' specialization for SAR test scenarios with 5-class target objects.

Agent ID#	**Robots Specialized Functionalities: 1 ≡ Possesses Functionality; 0 ≡ Does Not Possesses Functionality**					
	Specialty Vector	**Open Doors**	**Climb Stairs**	**Assist People**	**Extinguish Fire**	**Read Signs**
R_1	$\boldsymbol{S}_1$	1	1	0	0	0
R_2	$\boldsymbol{S}_2$	0	1	1	0	0
R_3	$\boldsymbol{S}_3$	0	1	0	1	0
R_4	$\boldsymbol{S}_4$	0	1	0	0	1
R_5	$\boldsymbol{S}_5$	0	0	1	0	0
R_6	$\boldsymbol{S}_6$	0	0	0	1	0
R_7	$\boldsymbol{S}_7$	0	0	0	0	1

5.2. Agents Fitting Probabilities Computation

The goal of the allocation scheme is to maximize the task-agent specialty fitting level defined as a probability. The estimated fitting probabilities of the individual swarm members are defined as:

$$\hat{\varphi}_{Ri} = \boldsymbol{S}_i \hat{\boldsymbol{P}}_T \tag{3}$$

where $\hat{\varphi}_{Ri} \in \mathcal{R}^{1\times 1}$ represents the estimated specialty collective score achieved by an individual agent, R_i, inferred from the confidence levels, $\hat{\boldsymbol{P}}_T \in \mathcal{R}^{F\times 1}$, on detected features on the target object, Equation (2). The fitting probabilities of Equation (3) are used to compute the swarms' cumulative probabilistic specialty fitting diagonal matrix, $\boldsymbol{Q} \in \mathcal{R}^{a\times a}$, that consists of the specialty fitting probabilities of all team members and is given as:

$$\boldsymbol{Q} = diag\begin{bmatrix} \frac{\hat{\varphi}_{R1}}{\varphi_{R_1}} & \frac{\hat{\varphi}_{R2}}{\varphi_{R_2}} & \cdots & \frac{\hat{\varphi}_{Ra}}{\varphi_{R_a}} \end{bmatrix} \tag{4}$$

$\varphi_{R_i} \in \mathcal{R}^{1\times 1}$, is the agent's maximum expected collective score that results when all of the agent's specialized capabilities are matched with their corresponding detected target. To define φ_{R_i}, the maximum number of specialized capabilities that are built in each individual agent are considered and φ_{R_i}, of agent R_i, can be defined as:

$$\varphi_{R_i} = \boldsymbol{S}_i \boldsymbol{p}_{max} \tag{5}$$

where,

$$\boldsymbol{p}_{max} = \left[p_{C_1 max}\, p_{C_2 max} \cdots p_{C_F max} \right]^T \tag{6}$$

$p_{C_i max}$ is the maximum expected confidence level on the detected target object for each class. As an example, based on object detection confidence levels shown in Table 2, the maximum expected detection score, $p_{C_i max}$, among all classes would be ≅ 0.995.

5.3. Qualified Agents Coordination

Beyond their specialty, the respective agents' availability information is also essential because an agent may not always be available when called in service. Therefore, the proposed scheme involves an agent's availability status, along with the agent's specialty fitting probabilities, $\boldsymbol{Q}$, Equation (4), for the coordination of qualified responders. As a result, the most qualified and available agent among the team is allocated to the detected task, even though it may not be the very best one (i.e., a less competent but available qualified agent at the moment of target object discovery may be selected). To provide this flexibility, an availability vector, $\boldsymbol{\vartheta}_{AS} \in \mathcal{R}^{a\times 1}$, is defined as a current internal state for each robot. At the time of swarm deployment, the internal flag of the deployed agents raises to "available", while the internal flag of agents that are not available is set to "withdrawn". Then, whenever the system finds an "available" agent that is qualified to allocate to a detected task, the availability state keeps the agent's specialty fitting probability active. The detected task is then assigned to the agent that is closer to the estimated location of the detected target object, provided that it is qualified to respond to the task. When an available and qualified agent is assigned to a given task, its availability state is changed to "busy", making this agent not available for any other assignment until completion of the current task assigned. On the other hand, the fitting probabilities of agents with an internal flag "withdrawn" or "busy" are deactivated, triggering the system to search for other "available" agents among the swarm. The availability vector of the team members, $\boldsymbol{\vartheta}_{AS} \in \mathcal{R}^{a\times 1}$, is defined as:

$$\boldsymbol{\vartheta}_{AS} = \begin{cases} \frac{1}{d_i} & \{R_i \textit{ is} \text{ "available" } \& \ d_i \ > \ r_{task} \\ 1 & \{R_i \textit{ is} \text{ "available" } \& \ d_i \ \leq \ r_{task} \\ 0 & \{R_i \textit{ is} \text{ "withdrawn" or "busy"} \end{cases} \tag{7}$$

d_i is the Euclidean distance between robot's R_i current location, (x_i, y_i), and the detected target location, (x_t, y_t), in the shared 2-D plane, and is given by:

$$d_i = | \sqrt{(x_i - x_t)^2 + (y_i - y_t)^2} | \tag{8}$$

Υ is a control variable that takes a binary value 1 or 0 to activate or eliminate the impact of the distance to the target's location; r_{task} is a predefined radius of the task zone that surrounds any detected target object [16]. Consequently, the coordination scheme is formulated as:

$$\boldsymbol{\Psi} = \boldsymbol{Q}\boldsymbol{\vartheta}_{AS} \tag{9}$$

where $\boldsymbol{\Psi} \in \mathcal{R}^{a\times 1}$ returns the fitting probabilities of the "available" robots, weighted by the inverse to their distance from the target, and 0's for the "withdrawn" and "busy" units, when a target object is recognized as a task to be performed with a related confidence level, $\hat{\boldsymbol{P}}_T$.

5.4. Human in the Loop

For increased safety and strategic management of the swarm's operation, a minimum task-agent fitting threshold (MFT), η, is also considered as a safety measure to guarantee a minimum level of qualification below which no agent will be allocated to any task. To adapt this parameter in a strategic manner according to operational conditions, a human operator is given access to the task allocation framework at a high level to supervise the swarm. This way, a provision is made for the human supervisor to share his skills with the robots and provide situational awareness, by dynamically adjusting the MFT that conditions the minimum expected confidence level on the recognition of target objects for the robotic agents to intervene.

The desired MFT, η, is selected by setting $\eta \in (0\ 1]$ over two predefined ranges: a low specialty fitting level (LSFL) and a high specialty fitting level (HSFL). The minimum limit of LSFL, $\eta \in (A\ B]$ drives the task-agent allocation scheme to match the very minimum specialized capabilities of the available agents to respond to the detected targets. However, in many applications, it is desired to ensure a higher level of confidence in the specialty-based task allocation to more selectively fit the capabilities of the available agents' with most of the requirements of the detected task. In such a case, the task allocator is enforced by the human supervisor to work in the HSFL range, $\eta \in (B\ C]$, by setting η above a specific level B to ensure that only robots with a higher level of competence can intervene, where:

$$\begin{cases} LSFL: & \mathrm{A} < \eta \leq \mathrm{B} \\ HSFL: & \mathrm{B} < \eta \leq \mathrm{C} \end{cases} \tag{10}$$

Therefore, $\mathbf{\Psi}$, defined in Equation (9), is further refined to only consider the probabilities of the available agents that achieve the desired MFT. The task allocation probabilities of the available responders, among the swarm of a agents, $\mathbf{\Psi}_{MFT} \in \mathcal{R}^{a\times1}$, are given by:

$$\mathbf{\Psi}_{MFT} = [\Psi_{MFT1},\ \Psi_{MFT2}, \cdots, \Psi_{MFTa}]^T \tag{11}$$

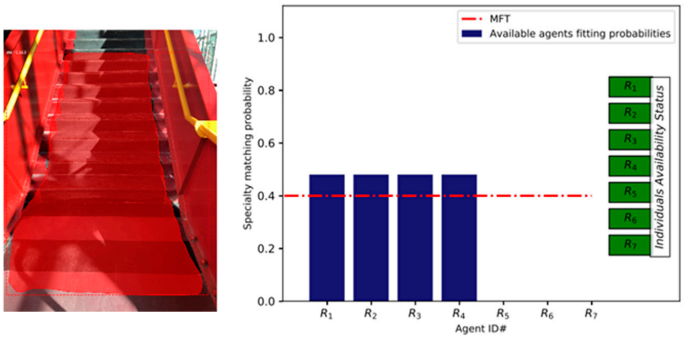
$$\text{where } \Psi_{MFTi} = \begin{cases} \Psi_i, & \big|\ \Psi_i \geq \eta\ :\ \Psi_i \in \mathbf{\Psi} \\ 0, & \big|\Psi_i < \eta\ :\ \Psi_i \in \mathbf{\Psi} \end{cases} \tag{12}$$

with $\{\Psi_i, i = 1, 2, \ldots, a\}$. Accordingly, the qualified available agents are automatically selected and allocated to the detected tasks considering the human's strategic guidance. For each detected target, the identification index, i, of the best-suited and available agent with a specialty fitting level above MFT among the swarm of robots, $\{R_i, i = 1, 2, \ldots, a\}$, is given by:

$$\varnothing_{BEST\ RESPONDER\ INDEX} = i \mid i\ \in max\{\mathbf{\Psi}_{MFT}\} \tag{13}$$

6. Experimental Results

A number of real test images were acquired with a camera while patrolling different sectors of a building with a ground mobile robot. Images were then processed to retrieve every instance of the five classes of target objects considered, as defined in Table 2. The maximum expected detection confidence, $p_{C_i max}$, among all classes is fixed to $\cong 0.995$. The robotic team is assumed to navigate on the ground floor of a building when the target object detection system recognizes a first instance of one of the predefined classes, e.g., stairs, as shown in Figure 4a. In this test case, the target objects are detected within the predefined task zone, which leads to, $\vartheta_{AS_i} = 1$, in Equation (7). The target object's detection confidence level is processed through the task allocation scheme to compute the individual robots' probabilistic fitting level, Equation (11), in order to assign the most qualified agents using Equation (13), to the detected task. Figure 4b shows that the confidence in robots R_1, R_2, R_3, R_4 being qualified to proceed and climb the stairs is beyond the desired MFT, while robots R_5, R_6, R_7 are not qualified. The robots' availability status is also presented in Figure 4b, with available agents shown as green squares and withdrawn agents as red squares. The detailed target detection confidence scores and the corresponding robots' task allocation fitting probabilities are reported in Table 4.



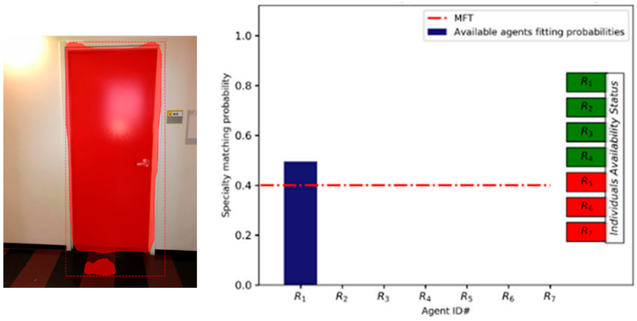
(a) **(b)**

Figure 4. (**a**) Detected stairs; (**b**) agents fitting probabilities, and agents' availability status (green = available).

Table 4. Swarm members fitting probabilities to climb stairs in SAR scenario in indoor workspace.

Target Objects Detection Confidence	**Robot ID#**	**Availability 1: Available 0: Withdrawn**	**Available Agents Fitting Probabilities**	**User Set MFT**
Door: 0.00 Stairs: 0.96 Person: 0.00 Fire: 0.00 Sign: 0.00	R_1	1	0.48	0.4
	R_2	1	0.48	
	R_3	1	0.48	
	R_4	1	0.48	
	R_5	1	0.00	
	R_6	1	0.00	
	R_7	1	0.00	

Next, the selected swarm members, R_1, R_2, R_3, R_4, get over the stairs and begin navigate the open space on the second floor. Then, a door is detected as shown in Figure 5a. The system computes the individual agents' specialty fitting probabilities, Equation (11), as shown in Figure 5b, to assign the most qualified agent, using Equation (13), to open the detected door. The availability state of the swarm members indicate that agents, R_1, R_2, R_3, R_4, are still available, whereas agents R_5, R_6, R_7, are withdrawn, as these agents were not qualified to initially climb the stairs and reach to the current task location corresponding to the detected door, which resulted in $\vartheta_{AS_{5,6,7}} = 0$, as defined in Equation (7). The results show that the fitting probability of agent R_1 with the detected door equals 0.49 (Table 5). As R_1, which is also the only agent with the capability to open a door, according to Table 3, has the highest fitting probability, which also exceeds the MFT, and is available, it is assigned to open the detected door.



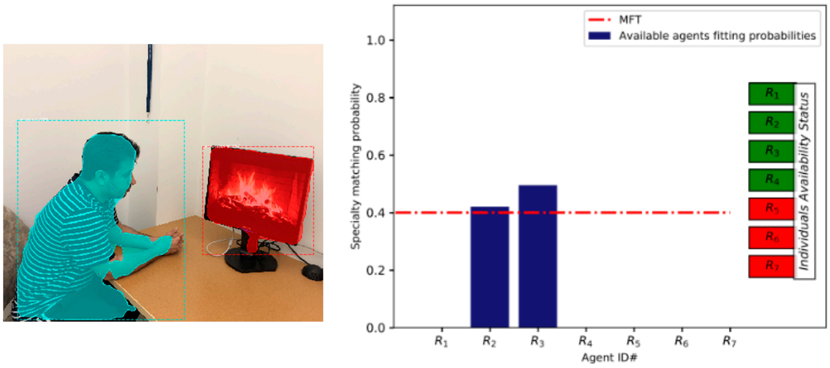
(a) (b)

Figure 5. (**a**) Detected target object (door); (**b**) specialized agents fitting probabilities, and agents' availability status (green = available; red = withdrawn).

Table 5. Individual agents' fitting probabilities to open a detected door in SAR scenario in indoor workspace.

Target Objects Detection Confidence	Robot ID#	Availability 1: Available 0: Withdrawn	Available Agents Fitting Probabilities	User Set MFT
Door: 0.98 Stairs: 0.00 Person: 0.00 Fire: 0.00 Sign: 0.00	R_1	1	0.49	0.4
	R_2	1	0.0	
	R_3	1	0.0	
	R_4	1	0.0	
	R_5	0	—	
	R_6	0	—	
	R_7	0	—	

Once the previously allocated robot,R_1, opens the door, then the remaining swarm members, R_1, R_2, R_3, R_4, access the workspace and the object detection stage conducts a new survey to detect additional target objects. A fire (tv-monitor) and a human victim in the vicinity of the fire are detected, as shown in Figure 6a. The detection results are leveraged by the task allocation scheme to determine the specialty fitting probabilities, Equation (11), among the still available agents, as shown in Figure 6b, and detailed in Table 6. The most competent and available agents, R_2, R_3, are assigned respectively, based on Equation (13), to each of the detected tasks.



(**a**) (**b**)

Figure 6. (**a**) Detected target objects: (left) person to be assisted, and (right) fire to be extinguished; (**b**) specialized agents' fitting probabilities, and agents' availability status.

Table 6. Agents fitting probabilities to respond to two simultaneously detected tasks in SAR scenario.

Target Objects Detection Confidence	Robot ID#	Availability 1: Available 0: Withdrawn	Available Agents Fitting Probabilities	User Set MFT
Door: 0.00 Stairs: 0.00 Person: 0.84 Fire: 0.99 Sign: 0.00	R_1	1	0.00	0.4
	R_2	1	0.42	
	R_3	1	0.49	
	R_4	1	0.00	
	R_5	0	—	
	R_6	0	—	
	R_7	0	—	

As a result, while guaranteeing a minimum confidence level (MFT) in the allocation process to ensure the safety of the operation, task allocation is successfully performed on unique or multiple detected targets throughout the scenario with the most qualified and available agents being automatically assigned as responders to the detected targets.

7. Quantitative Analysis of Performance

In order to generalize the evaluation of performance for the proposed integrated task allocation framework, Table 7 summarizes experimental results obtained for target object recognition over 140 captured images with instances of the five classes considered in the simulated SAR scenario. This test set contains images that were not considered as part of the training and validation datasets, detailed in Table 1. The target object detection overall precision over all classes is 92.9%, which indicates that the trained detection model can correctly recognize over 90% of object instances in these images (true positives), while the overall recall is 66.6%, indicating that over 30% of the instances failed to be detected (false negatives).

Table 7. Object recognition performance on captured images from a testing set.

	Person	Fire	Door	Sign	Stairs	Overall
Precision (%)	98.2	87.5	91.2	94.7	86.4	92.9
Recall (%)	81.2	45.5	67.4	66.7	95.0	66.6

The 140 test cases were considered to support task allocation for seven specialized robots as defined in Table 3. Over these test cases, the object recognition stage failed to recognize any object and resulted in no agent allocation in 12 cases (8.6%), similar to case 15 in Table 8. Additionally, out of the 140 test cases, 9 (6.4%) presented a misclassification error. For example, the lines on the floor are classified as stairs in case 8 of Table 8. In cases 5, 8, and 11 of Table 8, one of the detected targets is not allocated to an agent because of the low confidence level on the target object detection which is below the set MFT. Also, in cases 7 and 8 the last target is not allocated because all of the available corresponding specialized agents are busy with their allocation to another task. The proposed task allocator was successful in 93.6% of the trials to allocate proper agents to the detected corresponding targets. In all successful cases, the framework assigned the most specialized and available agents that achieved the minimum MFT on the probabilistic match between the available agent's specialized capabilities and the constraints imposed by the detected target. In situations where no objects were detected or a low confidence level on the target object detection was achieved, the correct response was to perform no allocation. This approach is also highly efficient computationally. When considered independently from the recognition stage, it took on average 0.078 s to allocate agents over all 140 test cases. Therefore, the task allocation framework brings no computational bottleneck, considering that object recognition running on GPUs necessitated 0.22 s per image to detect target objects.

Table 8. Sample images containing less confident targets' recognition among the five classes considered, and automatically assigned robotic agents for detected target(s) by the proposed approach. MFT = 0.4.

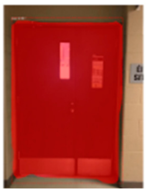
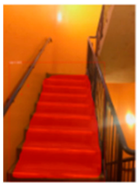
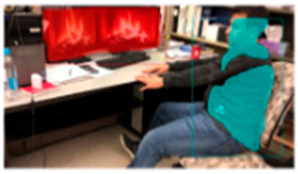
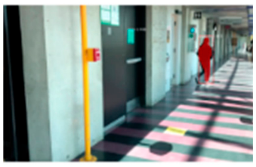
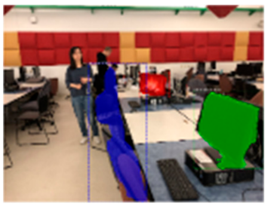
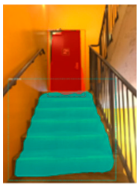
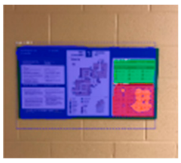
No.	Input Image with Segmented Detected Target(s)	Recognized Target Object Confidence Level		Assigned Agents	
				Agent	Fitting Probability
1		Door	0.991	R_1	0.49
		Stairs	0.000		
		Person	0.000		
		Fire	0.000		
		Sign	0.000		
2		Door	0.000		
		Stairs	0.965	R_1, R_2, R_3, R_4	0.48
		Person	0.000		
		Fire	0.000		
		Sign	0.000		

Table 8. *Cont.*

No.	Input Image with Segmented Detected Target(s)	Recognized Target Object Confidence Level		Assigned Agents	
				Agent	Fitting Probability
3		Door	0.000		
		Stairs	0.000		
		Person	0.858	R_5	0.86
		Fire	0.901	R_6	0.90
		Sign	0.000		
4		Door	0.000		
		Stairs	0.000		
		Person	0.987	R_5	0.99
		Fire	0.000		
		Sign	0.788	R_7	0.79
5		Door	0.000		
		Stairs	0.000		
		Person	0.985	R_5	0.98
		Fire	0.968; 0.628	R_6	0.97
				R_3 (not assigned)	0.31 < MFT
		Sign	0.000		
6		Door	0.847	R_1	0.42
		Stairs	0.978	R_2,R_3,R_4,	0.49
		Person	0.000		
		Fire	0.000		
		Sign	0.000		
7		Door	0.000		
		Stairs	0.000		
		Person	0.000		
		Fire	0.000		

		Sign	0.984;
			0.940;
			0.657
				R_7	0.98

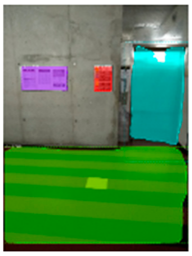
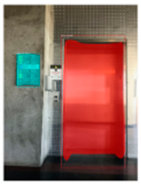
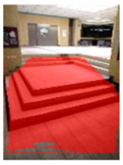
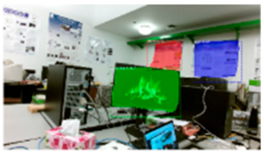
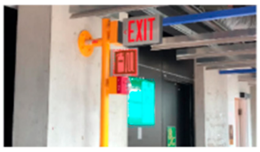
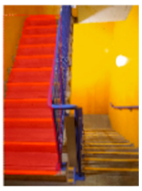
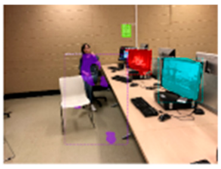
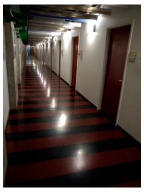
				R_4	0.47
				No specialized agent is available to allocate the third target	
8		Door	0.654	R_1	0.33 < MFT
		Stairs	0.970	R_1,R_2, R_3,R_4	0.49
		Person	0.000		
		Fire	0.000		
		Sign	0.995; 0.991	R_7	1
				No specialized agent is available to allocate the second target	
9		Door	0.993	R_1	0.49
		Stairs	0.000		
		Person	0.000		
		Fire	0.000		
		Sign	0.733	R_7	0.73

Table 8. *Cont.*

No.	Input Image with Segmented Detected Target(s)	Recognized Target Object Confidence Level		Assigned Agents	
				Agent	Fitting Probability
10		Door	0.000		
		Stairs	0.995	R_1, R_2, R_3, R_4	0.5
		Person	0.000		
		Fire	0.000		
		Sign	0.000		
11		Door	0.000		
		Stairs	0.000		
		Person	0.000		
		Fire	0.987	R_6	0.99
		Sign	0.843; 0.624	R_7	0.84
				R_4 (not assigned)	0.31 < MFT
12		Door	0.000		
		Stairs	0.000		
		Person	0.000		
		Fire	0.000		
		Sign	0.928, 0.824	R_7	0.93
				R_4	0.41
13		Door	0.000		
		Stairs	0.977	R_1, R_2, R_3, R_4	0.49
		Person	0.000		
		Fire	0.000		
		Sign	0.000		
14		Door	0.000		
		Stairs	0.000		
		Person	0.630	R_5	0.63
		Fire	0.913; 0.879	R_6	0.91
				R_3	0.44
		Sign	0.963	R_7	0.96
15		Door	0.000	——	
		Stairs	0.000		
		Person	0.000		
		Fire	0.000		
		Sign	0.000		

8. Comparison

Many factors are considered in this study to design a specialty-based task allocation approach that maximizes the task execution efficiency, and to expand the range of potential applications. The function considered here is to maximize a task-agent specialty fitting probability, while matching detected features on target objects with the respective robotic agents' specialized capabilities. In this section, the essence of the proposed approach is compared with four alternative task allocation mechanisms proposed in the literature for service and exploration robots. It highlights the main conceptual differences with previous literature and demonstrates how the original framework proposed and experimentally validated in this paper contributes an innovative path to address the task allocation problem in multi-robot systems.

8.1. Interface Delay Task Allocation (IDTA)

The task allocation approach presented in [24] partitions the foraging task into simpler subtasks called harvesting and storing subtasks. These two subtasks are sequentially inter-dependent, which means that the execution of one sub-task is conditioned by the execution of the other one. As a result, an item is transported from a source position to a task interface area by a harvesting agent. Next, the harvesting agent waits for an available agent that is involved in the storing subtask to deliver the item to that agent, which will pass it to the nest area. Similarly, a storing agent waits at the task interface border for an available agent that is engaged in the harvesting subtask to pick up the item. This task allocation technique is introduced based on a waiting time that is measured by the agents at the task's interface. It enables a swarm of service robots to dynamically partition the agents into two specialized groups. The individual agents work autonomously based on a decentralized control strategy, similar to the proposed approach in this paper. However, this task allocation scheme does not require the agents to communicate, whereas each individual agent switches between the harvesting and storing subtasks using the locally measured information about the time that the robot must wait to transfer the item at the task interface. The interface delay task allocation method might be an efficient approach to enable the robotic agents to move between two subtasks; however, it does not offer an efficient approach for a swarm that has a wide variety of functionalities involved in allocating tasks with different requirements that demand specific agents' functionalities. It also imposes the existence of a formal interface in between the agents where their role is transformed, a constraint that the proposed specialty-based task allocation scheme does not bring into the formulation, therefore providing superior flexibility into the definition of tasks and the freedom of movement for every agent.

8.2. Multiple Travelling Salesman Assignment (MTSA)

This task allocation approach selects the next navigational goal using the famous travelling salesman problem (TSP) distance cost [25]. The latter is defined as the travelled distance on the shortest path that connects the robot position with the candidate goals. This task allocation mechanism is developed for a single robot exploration that navigates many goal points and from which the exploration mission can cover all frontier cells. This task allocation approach is optimal for a single robot mission to perform exploration tasks; however, the problem of computing the optimal distance between the robot position and a set of goals only considers the shortest travelling distance. In comparison, the proposed specialty-based task allocation method deals with an indefinite and flexible number of agents; it optimizes the selection of agents beyond just the travelling distance; and it easily adapts to a wide range of robot's specialization considerations according to the nature of the tasks to be performed and the type of physical resources involved in addressing a situation. Moreover, it allows strategic input and guidance from a human supervisor when needed, while a travelling salesman optimization approach does not offer such a flexibility.

8.3. Taxonomy of Multi-Agent Task Allocation (An Optimization Approach)

A formal taxonomy of multi-robot task allocation problems is introduced in [26]. This study classifies previous solutions for multi-robot task allocation problems based on an optimization theory. The authors of this work propose an architecture-independent taxonomy with the goal to optimize task allocation. The problem is addressed at three levels: First, the robot level, which captures the capability for a robot to execute either a single task or multiple tasks. Second, the task level, which defines whether the task requires a single robot or multiple robots to be completed. Third, the task allocation time level which determines whether the task should be executed instantaneously with no planning for future assignments, or a set of tasks should be assigned over time. Finally, task allocation is processed as an optimization approach to improve the performance of the system while assuming that each robot can estimate its capability to perform each task based on two factors: 1) the task execution quality, and 2) the expected cost in terms of resources. The formulation is general and can adapt to a variety of application contexts. However, the solution does not construct a formal model to capture the agents' heterogeneous functionalities, formulated as specializations in our work, to be formally matched with explicit constraints monitored on the task to be performed.

8.4. Task-Allocation Algorithms in Multi-Robot Exploration

The multi-robot task allocation problem is also investigated in [27] to allocate navigational goals to multiple robots in exploration tasks. In this work, the task allocation problem is addressed as a classical distance cost and the proposed approach essentially guides the robot to the nearest navigational goal. However, a formal correspondence of the task constraints and the resources available on the robotic agents is not considered in this approach.

9. Conclusions

The design of a formal representation for specializing individuals of a robotic swarm and forming an association with corresponding characteristics on visually detected target objects is introduced in this paper. A target object detection using Mask R-CNN technique is integrated with the proposed task allocation approach. The framework is validated with real images collected in indoor environments and involving simulated mobile robot navigation scenarios. The specialized capabilities of individual robotic agents are modeled and matched to corresponding visual features recognized on target objects with a quantified confidence level. That confidence level is associated with specific task requirements and is used to tune the task-agent probabilistic matching scheme. Specialized individual agents are coordinated with corresponding tasks while considering the agents availability state along with their probabilistic specialty fitting level. The framework also supports strategic guidance from a human operator to refine the task assignment process with situational awareness. The process is designed to keep human's cognitive load low while adjusting the system's operational conditions at a high level of coordination only, which results in safer and more selective task allocation operation. Experimental results demonstrate that the proposed approach is successful at properly assigning specialized agents to corresponding tasks that require specific mechanical or instrumentation characteristics from autonomous robots. Future developments of the proposed framework will encode the agents' specialization vector in a non-binary form to modulate the agents' specialized functionalities based on the robustness of their hardware and software implementation and to capture different levels of suitability of the specializations to different tasks.

Author Contributions: O.A.-B. contributed to the development of the overall framework in terms of conceptualization, methodology formal analysis, and original draft preparation. W.W. contributed to the development of the target object recognition stage. P.P. supervised and administered the project. All authors have read and agreed to the published version of the manuscript.

Funding: The authors wish to acknowledge the support from Department of National Defence of Canada toward this research under the Innovation for Defence Excellence and Security (IDEaS) program, CP_0622, as well as support from Hadramout Foundation.

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

References

1. Korte, B.; Vygen, J. *Combinatorial Optimization: Theory and Algorithms*; Springer: Berlin, Germany, 2008.
2. Hall, P. On representatives of subsets. In *Classic Papers in Combinatorics*; Birkhäuser: Boston, MA, USA, 2009; pp. 58–62.
3. Jones, C.; Mataric, M.J. Adaptive Division of Labor in Large-scale Minimalist Multi-robot Systems. In Proceedings of the 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems, Las Vegas, NV, USA, 27–31 October 2003; Volume 2, pp. 1969–1974.

4. Smith, S.L.; Bullo, F. Target assignment for robotic networks: Asymptotic performance under limited communication. In Proceedings of the 2007 American Control Conference, New York, NY, USA, 9–13 July 2007; pp. 1155–1160.
5. Claes, D.; Robbel, P.; Oliehoek, F.A.; Tuyls, K.; Hennes, D.; van der Hoek, W. Affective Approximation for Multi-Robot Coordination in Spatially Distributed Tasks. In Proceedings of the Intl Conference on Autonomous Agents and Multi-agent Systems, Istanbul, Turkey, May 2015; 2015; pp. 881–890.
6. Yasuda, T.; Kage, K.; Ohkura, K. Response Threshold-Based Task Allocation in a Reinforcement Learning Robotic Swarm. In Proceedings of the IEEE 7th International Workshop on Computational Intelligence and Applications (IWCIA), Hiroshima, Japan, 7–8 November 2014; pp. 189–194.
7. Wu, H.; Li, H.; Xiao, R.; Liu, J. Modeling and simulation of dynamic ant colony's labor division for task allocation of UAV swarm. *Phys. A Stat. Mech. Appl.* **2018**, *491*, 127–141. [CrossRef]
8. Matarić, M.; Sukhatme, G.; Qstergaard, E. Multi-Robot Task Allocation in Uncertain Environments. *Auton. Robot.* **2003**, *14*, 255–263. [CrossRef]
9. Amigoni, F.; Brandolini, A.; Caglioti, V.; Di Lecce, C.; Guerriero, A.; Lazzaroni, M.; Lombardo, F.; Ottoboni, R.; Pasero, E.; Piuri, V.; et al. Agencies for perception in environmental monitoring. *IEEE Trans. Instrum. Meas.* **2006**, *55*, 1038–1050. [CrossRef]
10. Rastegari, M.; Ordonez, V.; Redmon, J.; Farhadi, A. XNOR-Net: Imagenet classification using binary convolutional neural networks. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2016; pp. 525–542.
11. Lins, R.G.; Givigi, S.N.; Kurka, P.G. Vision-based measurement for localization of objects in 3-D for robotic applications. *IEEE Trans. Instrum. Meas.* **2015**, *64*, 2950–2958. [CrossRef]
12. Wu, W.; Payeur, P.; Al-Buraiki, O.; Ross, M. Vision-Based Target Objects Recognition and Segmentation for Unmanned Systems Task Allocation. In Proceedings of the International Conference on Image Analysis and Recognition, Waterloo, ON, Canada, 27–30 August 2019; Karray, F., Campilho, A., Yu, A., Eds.; Springer: Cham, Switzerland, 2019; pp. 252–263.
13. Ren, S.; He, K.; Girshik, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada, 7–12 December 2015; pp. 91–99.
14. He, K.; Gkioxari, G.; Dollar, P.; Girshick, R. Mask R-CNN. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017.
15. Al-Buraiki, O.; Payeur, P.; Castillo, Y.R. Task switching for specialized mobile robots working in cooperative formation. In Proceedings of the IEEE International Symposium on Robotics and Intelligent Sensors, Tokyo, Japan, 17–20 December 2016; pp. 207–212.
16. Al-Buraiki, O.; Payeur, P. Agent-Task assignation based on target characteristics for a swarm of specialized agents. In Proceedings of the 13th Annual IEEE International Systems Conference, Orlando, FL, USA, 8–11 April 2019; pp. 268–275.
17. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.
18. Bashiri, F.S.; LaRose, E.; Peissig, P.; Tafti, A.P. MCIndoor20000: A fully-labeled image dataset to advance indoor objects detection. *Data Brief* **2018**, *17*, 71–75. [CrossRef] [PubMed]
19. Everingham, M.; van Gool, L.; Williams, C.K.I.; Winn, J.; Zisserman, A. The PASCAL Visual Object Classes (VOC) Challenge. *Int. J. Comput. Vis.* **2010**, *88*, 303–338. [CrossRef]
20. Russell, B.C.; Torralba, A.; Murphy, K.P.; Freeman, W.T. LabelMe: A Database and Web-based Tool for Image Annotation. *Int. J. Comput. Vis.* **2008**, *77*, 157–173. [CrossRef]
21. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
22. Lin, T.Y.; Dollar, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 936–944.
23. Lin, T.-Y.; Maire, M.; Belongie, S.; Hayes, J.; Perona, P.; Ramanan, D.; Dollar, P.; Zitnick, C.L. Microsoft COCO: Common Objects in Context. In *13th European Conference on Computer Vision, Zurich, Switzerland, LNCS*; Springer: Cham, Switzerland, 2014; Volume 8693, pp. 740–755.

24. Brutschy, A.; Pini, G.; Pinciroli, C.; Birattari, M.; Dorigo, M. Self-organized task allocation to sequentially interdependent tasks in swarm robotics. *Auton. Agents Multi-Agent Syst.* **2014**, *28*, 101–125. [CrossRef]
25. Kulich, M.; Faigl, J.; Přeučil, L. On distance utility in the exploration task. In Proceedings of the IEEE International Conference on Robotics and Automation, Shanghai, China, 9–13 May 2011; pp. 4455–4460.
26. Gerkey, B.; Matarić, M. A formal analysis and taxonomy of task allocation in multi-robot systems. *Int. J. Robot. Res.* **2004**, *23*, 939–954. [CrossRef]
27. Faigl, J.; Olivier, S.; Francois, C. Comparison of task-allocation algorithms in frontier-based multi-robot exploration. In *European Conference on Multi-Agent Systems*; Springer: Cham, Switzerland, 2014; pp. 101–110.

© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).