

# Visual victim detection and quadrotor-swarm coordination control in search and rescue environment

Gustavo A. Cardona<sup>1</sup>, Juan Ramirez-Rugeles<sup>2</sup>, Eduardo Mojica-Nava<sup>3</sup>, Juan M. Calderon<sup>4</sup>

<sup>1,2,3</sup>Department of Electrical and Electronic Engineering, Universidad Nacional de Colombia, Colombia

<sup>1</sup>Department of Computer Science and Engineering, Bethune Cookman University, Daytona, Florida

<sup>4</sup>Department of Electronic Engineering, Universidad Santo Tomas, Colombia

---

## Article Info

### Article history:

Received Jul 11, 2020

Revised Dec 21, 2020

Accepted Jan 5, 2021

### Keywords:

Consensus

Convolutional  
neural-networks

Quadrotors

Swarm-navigation

Victim-detection

## ABSTRACT

We propose a distributed victim-detection algorithm through visual information on quadrotors using convolutional neuronal networks (CNN) in a search and rescue environment. Describing the navigation algorithm, which allows quadrotors to avoid collisions. Secondly, when one quadrotor detects a possible victim, it causes its closest neighbors to disconnect from the main swarm and form a new sub-swarm around the victim, which validates the victim's status. Thus, a formation control that permits to acquire information is performed based on the well-known rendezvous consensus algorithm. Finally, images are processed using CNN identifying potential victims in the area. Given the uncertainty of the victim detection measurement among quadrotors' cameras in the image processing, estimation consensus (EC) and max-estimation consensus (M-EC) algorithms are proposed focusing on agreeing over the victim detection estimation. We illustrate that M-EC delivers better results than EC in scenarios with poor visibility and uncertainty produced by fire and smoke. The algorithm proves that distributed fashion can obtain a more accurate result in decision-making on whether or not there is a victim, showing robustness under uncertainties and wrong measurements in comparison when a single quadrotor performs the mission. The well-functioning of the algorithm is evaluated by carrying out a simulation using V-Rep.

*This is an open access article under the [CC BY-SA](#) license.*



---

## Corresponding Author:

Gustavo A. Cardona

Department of Electrical and Electronic Engineering

Universidad Nacional de Colombia

Cra 45, Bogotá, Colombia

Email: gacardonac@unal.edu.co

---

## 1. INTRODUCTION

Nowadays, one of the main areas in which the robotics research community is working is in search and rescue (SAR) missions assistance by the use of mobile robots in disaster zones to safeguard as many lives as possible. Earthquakes, floods, hurricanes, fires, are just some of the most frequent scenarios that put human lives in endangered. On one hand, the current advance of technology has allowed robots to have the highest apogee as a possible solution or improvement to the SAR tasks, there are many fields where robotics could intervene optimizing the performance of some human SAR tasks, such as mapping of the environment, detection of victims, and deployment of first aids. On the other hand, the development of robotics theory has improved the algorithms applied to robots in natural disaster zones increasing the probability of finding survivors, as observed in [1]. In the same way, in [2] it is explained why it is vital to act on the affected

area within 48 hours of the incident due to the probability of finding survivors gets significantly reduced after that period. Bearing in mind that time is crucial, applying robotics to SAR missions seems to be advantageous, relying on the robots' ability to carry out tasks efficiently in conditions that might be adverse for human rescuers as presented in [3]. Likewise, there are works as [4, 5] that show the possibility of employing robot swarms capable of behave cooperatively with humans improving the tasks performance.

Particularly in the robotics SAR area, it is possible to perform the exploration and mapping tasks of the affected area employing robots in a shorter time than carried out by a conventional rescue team [6], allowing to detect victims and navigate within the area faster. Regardless of several works are tackling the problem, there are still multiple challenges to be solved, such as navigation through non-convex spaces, the implementation of robust victim detection and estimation algorithms, the distributed SLAM, among others. Some of the most popular techniques that have been used in these tasks are bio-inspired or vision-based approaches as [7–9]. However real disaster environments are difficult to get access due to usually those are unpredictable and dangerous making difficult to test algorithms. In contrast, virtual environments are really easy to be accessed thanks to the development of simulator motors such as V-Rep and Gazebo, these must capture as many features as possible from real disaster environments, allowing algorithms to be tested and evaluated as shown in [2]. Another important task that robots can do in a SAR mission is victim detection due to the number of robots increases the resiliency and robustness of the detection in comparison to the human rescuers that usually are available. Here is important to consider that the algorithms implemented allow diminishing the effects of the gap in visual information in virtual and real environments sensed by the robot cameras.

The difficulty level of finding victims in a SAR scenario is increasing when the coordination and synchronization of a swarm-robotics are considered as in [10], as well the coordination needed when the victims may move inside the risk zone, is important to be able to track the dynamic targets and maintain communication within the swarm, like the task allocation consensus algorithm developed in [11]. other ideas giving the popularity in drone technology is the use of this drones to cover areas of interest such as crowded areas where a lot of mobile devices can provided rich information or the drones can be deploy to monitor climate changes anywhere anytime, like seen in [12, 13] respectively. Regarding only the detection process, there are multiple ways to proceed using different types of sensors, such as radar or cameras. When using radar systems, aspects such as the speed of processing, the instrumental accuracy and the need of sophisticated algorithms are relevant to achieve an acceptable operation under adverse conditions [9, 14]. On the contrary, a camera sensor as in [7] allows the system to acquire a lot of information from the environment, challenge here becomes to detect people in images taking into account that not all of the victims in a catastrophe are in the same position, neither they have similar features, as shape, clothes color, size, rotation, and occlusion. Regardless of the sensor used, one of the approaches that have become relevant to read the sensor information and interpret is convolutional neural networks (CNN). The CNN technique derived from learning algorithms and artificial intelligence is a tool that permits the system to be trained and recognize objects in an image by finding target features. Once the CNN is trained the sensor takes information about the environment and recognizes relevant objects that allow the system to develop behaviors according to the information perceived as shown in [15].

The challenge in the development of control algorithms based on well-functioning neural networks, lies in terms of computational expense and processing time, so that the actions are executed in the shortest possible time and in a correct way [16]. Other authors have worked on victim identification in SAR environments, for instance in [8], histograms of oriented gradients, based on color skin detection which could be sensitive to lightening conditions. In [17] the problem of light conditions is tackled generating a robust Viola-Jones algorithm that detect victim humans. In the same way, [18] use a transformation space through Gabor approach to detect humans based on skin color on a RGB image processed by photogram captured in a video. In contrast, our approach focuses in a robust way to detect a victim based on many features that a CNN can choose. Considering that SAR environments have highly exposure to occlusion and also that human skin might be altered by other components in the environment. On the other hand, there are some other works that works with sophisticated robots such as [19] where a robot is capable to identify a victim by the use of a infrared camera and a lidar sensor. In spite of the fact it is necessary to have redundancy in the sensors and determine whether or not the robot is detecting a victim this robot can be expensive and since robots are exposed to drop out due to attrition, it might be better to consider a cheaper robot but deploying multiple of them providing greater robustness to the system. We assumed the communication through the multi-agent network and the base system is done. Our research approach is not focused on the communication issue as it was dealt with in [20, 21]. Instead of it, we are focused on the robot swarm navigation [22] and victim detection through multi-agent consensus.

In this paper we extend the work [23] where an aerial multi-quadrotor platform capable to navigate in a virtual SAR environment with obstacles in it is carried out. The contribution of this paper is threefold. First, we consider the non-linear dynamic of the quadrotor instead of the linear which allows the algorithm to generate non-smooth trajectories. Second, we consider the use of cameras in each quadrotor to acquire virtual images from V-Rep simulator with the inclusion of fire that occludes the victim. Finally, we prove that the max estimation consensus, brings a better outcome than the well-known consensus algorithm when there is occlusion in the images. We make use of the camera to acquire information of the virtual environment. A data-set of 25.000 pictures was used in order to train a CNN to allow the system to identify victims in the virtual environment and then give to each robot in the swarm the trained CNN. Once that each robot in the swarm is capable of detecting victims, when some of them detect a victim in the same place they disrupt their communication to the main swarm and generate new links to a sub-swarm. These sub-swarms let the main swarm keep navigating while they perform a consensus formation algorithm in order to navigate around the victim while at the same time performing either a estimation consensus (EC) or a max estimation consensus (M-EC) to determine if there is or not a victim, making the system robust in contrast when just one robot tries to detect the victim.

The other sections that completes this paper are organized as follows: Section 2, swarm navigation and consensus algorithms, here is shown the quadrotors model, and how they navigate in the environment until they detect a possible victim, where the generation of sub-swarm are carried out, and its formation control applied around the victim. Section 3, visual victim detection, show how the CNN in each quadrotor identifies the victim through its camera, and then an estimation consensus. Section 4, shows some simulation and results that validate the algorithms. Finally, section 5 corresponds to the document conclusions and some future work alternatives.

## 2. SWARM NAVIGATION AND CONSENSUS ALGORITHM

We consider a set of quadrotors  $\mathcal{N} = \{1, 2, \dots, n\}$  whose interaction are modeled via graph  $\mathcal{G} = (\mathcal{N}, \mathcal{E})$ , where  $\mathcal{E}$  represents the communication between quadrotors. Each quadrotor  $i \in \mathcal{N}$  has a corresponding state variable  $x_i \in \mathbb{R}^3$ , which are the location of the quadrotor regarding to each axis of the space  $\mathcal{S} \in \mathbb{R}^3$ . It is necessary to state that,  $\mathcal{S}$  is a non-convex space which means that is composed by either areas clear to move by the quadrotors  $\mathcal{S}_f \in \mathbb{R}^3$  and areas with obstacles  $\mathcal{S}_o \in \mathbb{R}^3$  that disallow quadrotors to go thru, noting that  $\mathcal{S} = \mathcal{S}_f \cup \mathcal{S}_o$ .

### 2.1. Robot swarm navigation

The movement of the quadrotors is similar to the one shown in [23] where each quadrotor target position is determined by a single integrator dynamic  $\dot{x}_i^d = u_i$ , where  $\dot{x}_i^d$  are the linear target velocities of each quadrotor and  $u_i$  is the signal control to be designed. The approach used to give the desired target position to quadrotors is artificial potential functions, which emulates the attraction and repulsion behaviors presented in nature as in Reynolds rules. Allowing quadrotor  $i \in \mathcal{N}$  to maintain a comfortable distance to obstacles and neighbor quadrotors  $j \in \mathcal{N}_i$ , where  $\mathcal{N}_i$  is the neighborhood of quadrotor  $i^{th}$ . The control signal is a summation of both attraction and repulsive forces as,

$$u_i = u_{a_i} + u_{r_i} + u_{o_i},$$

where,  $u_{a_i} = -k_{a_i}(x_i^d - x_j^d)$  is the attraction force,  $k_{a_i} \in \mathbb{R}_{>0}$  is an established constant. On the other hand, the repulsion force when a comfortable distance is reached is defined as,  $u_{r_i} = (-k_{r_i}(\|x_i^d - x_j^d\| - \Delta))(x_i^d - x_j^d)$ , in which,  $k_{r_i} \in \mathbb{R}_{>0}$  is an established constant,  $\Delta \in \mathbb{R}_{>0}$  is a minimum distance allowed between quadrotors and  $\|x_i^d - x_j^d\|$  is the euclidean distance in  $\mathbb{R}^3$ . In addition, it is necessary to avoid obstacles  $x_o = \{x \in \mathbb{R}^3 \mid x \in \mathcal{S}_o\}$  in the environment so another repulsion force is needed,  $u_{o_i} = k_{o_i} \exp((-1/2\|x_i^d - x_o\|^2)/r_s^2)(x_i^d - x_o)$ , where  $k_{o_i} \in \mathbb{R}_{>0}$ , is an established repulsion constant, and  $r_s \in \mathbb{R}_{>0}$  is the security radius in which the quadrotor avoid collisions and depend on the obstacle size. Considering the generation of desired locations to reach, the objective becomes to navigate in a known space being attracted to interesting points while avoiding collisions to both other quadrotors and obstacles. The complete swarm navigation behavior is shown in Figure 1, where the swarm navigates keeping the distance among quadrotors and avoiding obstacles at the same time.



Figure 1. Robot swarm navigation, (a) Robot swarm navigation avoiding a big obstacle, and (b) Robot swarm navigation avoiding multiple obstacles

Quadrotors dynamic is modeled by the use of Newton-Euler approach, based on [24, 25] in which the motion equations can be written as (1),

$$\begin{aligned}
 \dot{x}_i &= v_i, \\
 m_{q_i} \dot{v}_i &= m_{q_i} g(-Z_w) + F_i(Z_{B_i}), \\
 \dot{R}_i &= R_i \hat{\Omega}_i, \\
 J_i \dot{\Omega}_i &= -\Omega_i \times J_i \Omega_i + M_i,
 \end{aligned} \tag{1}$$

where  $\{X_w, Y_w, Z_w\}$  are unit vectors along the axes of  $\{\mathcal{W}\}$  which is the inertial reference frame,  $\{X_{B_i}, Y_{B_i}, Z_{B_i}\}$  unit vectors along the axes of the  $i^{th}$  quadrotor  $\{\mathcal{B}_i\}$  with respect to  $\{\mathcal{W}\}$ ,  $v_i \in \mathbb{R}^3$  is the linear velocities of the  $i^{th}$  quadrotor,  $m_{q_i}$  is the mass of each quadrotor,  $g$  represents the gravity,  $J_i \in \mathbb{R}^{3 \times 3}$  is the inertia matrix of the  $i^{th}$  quadrotor with respect to  $\{\mathcal{B}_i\}$ ,  $R_i \in \mathbb{R}^{3 \times 3}$  is the rotation matrix that relates  $\{\mathcal{B}_i\}$  with respect to  $\{\mathcal{W}\}$ ,  $\Omega_i \in \mathbb{R}^3$  angular velocity of  $i^{th}$  quadrotor in  $\{\mathcal{B}_i\}$ ,  $F_i \in \mathbb{R}$  total thrust produced by the  $i^{th}$  quadrotor  $M_i \in \mathbb{R}^3$  moment produced by the  $i^{th}$  quadrotor, and the hat map  $\hat{\cdot}: \mathbb{R}^3 \rightarrow \text{SO}(3)$  is the skew-symmetric operator matrix as explained in [26] such that,  $\hat{x}y - x \times y \quad \forall x, y \in \mathbb{R}^3$ . In (1) it is noticeable that the control inputs are  $F_i$  and  $M_i$ , and the control laws are found through the use of geometric control depicted in [24], where,  $F_i = F_i^{des} \cdot Z_{B_i}$ , controls the altitude dynamics in which  $F_i^{des} = -K_p e_{p_i} - K_v e_{v_i} + m_{q_i}(gZ_w + \ddot{x}_i^d)$ ,  $e_{p_i} = x_i - x_i^d$  the position error,  $e_{v_i} = v_i - \dot{x}_i^d$  the velocity error, and  $k_p, K_v \in \mathbb{R}_{>0}$  are proportional gains. On the other hand, the attitude dynamics is controlled by  $M_i = -K_R e_{R_i} - K_\Omega e_{\Omega_i} + (\Omega_i \times J_i \Omega_i) - J_i(\hat{\Omega}_i R_i^\top R_i^d \Omega_i^d - R_i^\top R_i^d \hat{\Omega}_i^d)$ , where  $a^\top$  is the transpose of the matrix  $a$ ,  $e_{R_i} = \frac{1}{2} ((R_i^{des})^\top R_i - R_i^\top R_i^{des})^\vee$ , is the rotation matrix error, the vee map  $\vee: \text{SO}(3) \rightarrow \mathbb{R}^3$  which is the inverse of the hat map,  $e_{\Omega_i} = \Omega_i - R_i^\top R_i^d \Omega_i^d$  in the angular velocity error,  $K_R, K_\Omega \in \mathbb{R}_{>0}$  are proportional gains. The control inputs guarantees that the quadrotor position tends to the desired position  $x \rightarrow x^d$  and proof is shown in [27].

## 2.2. Sub-swarm formation around a possible victim

Once the navigation avoiding collisions and maintaining connectivity is guaranteed, the next important behavior is to create sub-swarm when a victim in the environment is found. When at least one of the quadrotors detects a victim, it hovers over it, affecting the complete behavior of the swarm, reason why the robot has to break communication links with the rest of the quadrotors. Allowing the main swarm to move freely leaving a reduced group of quadrotors out of the graph. The way the quadrotors decide to leave the main swarm is through the use of K – nearest neighbors approach. The K – nearest algorithm behaves as a classifier by selecting as its name indicates the  $k$  closer quadrotors to the first quadrotor that detected a possible victim. When the neighborhood has been chosen as  $\mathcal{N}_{ss}$ , we create a sub-graph  $\mathcal{G}_{ss} \subset \mathcal{G}$ , where  $\mathcal{G}_{ss} = (\mathcal{N}_{ss}, \mathcal{E}_{ss})$ , which will be disconnected from the main graph  $\mathcal{G}$ . Here  $\mathcal{E}_{ss}$  is generated taking into account a weighted function  $\mathcal{W}_{ss}: \mathcal{E}_{ss} \rightarrow \mathbb{R}$  in the following way,  $\mathcal{W}_{ss} = 1/(x_f^d - x_j^d)$ , where  $x_f^d$  is the first quadrotor that detects the victim and  $x_j^d$  are the neighbors. Allowing in this way, that the closer quadrotors have more relevance to the classification of quadrotor  $i^{th}$  between take itself to the sub-swarm or remain in the main swarm.

### 2.3. Formation control

When the sub-swarm is achieved all of the quadrotors that belong to the sub-swarm start to sense the victim through its cameras. Each quadrotor obtains an own percentage value of the victim recognition all the time, here we perform a formation control based on consensus, which gives quadrotors the possibility to acquire as much information as possible about the victim, from different perspectives allowing them to determine more precisely if it is a victim or not. Hence, the desired formation control considers the following dynamics just for the quadrotors that belong to the sub-swarm,  $\dot{x}_{ss_i}^d = -\sum_{j \in \mathcal{N}_{ss}} \nabla_{x_i^d} \psi_{ij}$ , where  $\psi_{ij}$  are the potential functions that guarantee to maintain connectivity, and avoid collisions while achieving a formation established. These potential functions are defined as,  $\psi_{ij} = \left(1/(\rho_2^2 - \|x_{ss_i}^d - x_{ss_j}^d\|^2)\right) - \left(1/(\|x_{ss_i}^d - x_{ss_j}^d\|^2 - \rho_1^2)\right)$ , in which  $\rho_2$  corresponds to the radius of connectivity and  $\rho_1$  the minimum radius to avoid collisions. The formation performed is shown in Figure 2, here the objective is to locate all the quadrotors equally distributed around the victim.

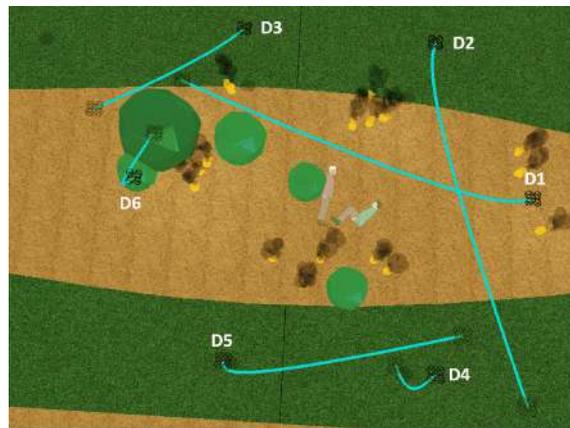


Figure 2. Path followed by the six quadrotors in the formation consensus

## 3. VISUAL VICTIM DETECTION

In order to perform the task of saving victims from a disaster zone not only the navigation algorithm is important but also the system that allows the robot to detect and localize victims. Thus, for each agent is important to recognize its environment and detect when a victim is in the near area. Each robot performs the detection task through visual information analysis using CNN. The image acquisition is based on use of robot's cameras which are the main sensors on this approach.

Once the robots who conform the sub-swarm are determined, the consensus formation is performed until every quadrotor reach its final position. During the time that formation consensus is performed, every quadrotor is using CNN to determine the certainty value of victim detection individually. While each quadrotor belonging to the sub-swarm determines the victim estimation measurement. The EC algorithm is applied using those values to calculate a concerted level victim detection. At the same time, the sub-swarm will provide more accurate information to the rescuers about the existence or not of a victim in the near zone. In this order of ideas some concepts of CNN and the sensing consensus are briefly described.

### 3.1. Basic concepts of convolutional neural-networks

CNN is a special class of artificial neural network (ANN) originally proposed in [28] that is used to process digital images in classification or identification tasks. These networks employ different convolutional filters with linear rectifiers intended to extract multiple interest features from the image as borders, corners or specific shapes. After the convolutional filtering, the resulting images are down-sampled in the so-called pooling process, reducing the size of the image but preserving most of the relevant information. Those two steps (convolution and pooling) are repeated several times, where every time, the result is a larger number of images with smaller dimensions. Finally, the values of the resulting images are given as input to a traditional neural network with fully connected layers whose weights are adjusted in a supervised training process feed by a large number of images properly labeled according to a human victim detection goal as shown in Figure 3.

For the case of victim detection, CNN has a single output in charge of determining if a victim was detected or not.

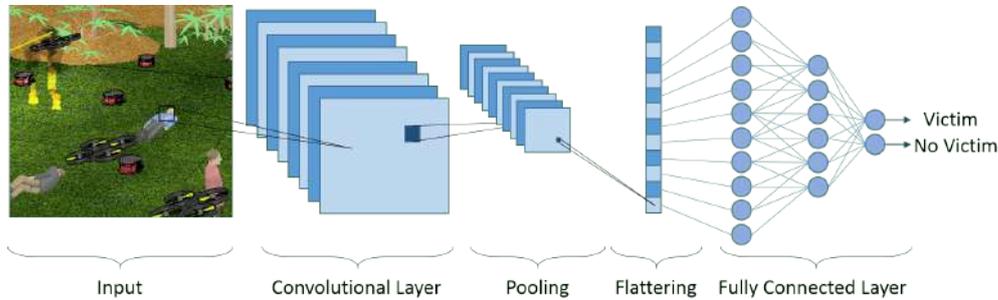


Figure 3. Digraph scheme

### 3.2. Consensus applied to a multi-sensor network for victim identification

When the sub-swarm that wants to identify the victim is determined as shown in section 2.2 a distributed estimation consensus will guarantee the robustness to uncertainties or malfunctioning of any sensor or quadrotor. The model of the estimation contemplated is based on distributed linear least square in presence of uncertainties as follows,  $\sigma_i = \mathcal{H}_i\beta + \varepsilon_i$ , where  $\beta$  is the estimation function, affected by uncertainties  $\varepsilon_i$ ,  $\sigma_i$  is the measurement channel of the  $i^{th}$  sensor, and  $\mathcal{H}$  is a variable that assures that the measurements are not entirely redundant. Taking into account that the aim of the least squares is to minimize the error, by isolating  $\varepsilon_i^2$  as  $\varepsilon_i^2 = \varepsilon_i\varepsilon_i^\top = (\sigma_i - \mathcal{H}_i\beta_i)^\top (\sigma_i - \mathcal{H}_i\beta_i)$ , the outcome is a function  $f(\beta_i)$  which depends only of  $\beta_i$ . The purposes of this algorithm can be then described by the following minimization,

$$\min \sum_{i=1}^n f_i(\beta) \\ s.t. \beta \in \mathbb{R}^q$$

in which  $f_i: \mathbb{R}^q \rightarrow \mathbb{R}$  are convex functions, as a consequence the optimal point is found in the average of the gradient functions of each sensor as  $f^* = \frac{1}{n} \sum_{j=1}^n (f_j)$ , additionally considering the minimum is given in,

$$\hat{\beta}_i = \left( \sum_{i=1}^n \mathcal{H}_i^\top \mathcal{H}_i \right)^{-1} \left( \sum_{i=1}^n \mathcal{H}_i^\top \sigma_i \right),$$

then the distributed estimation will have form (2),

$$\hat{\beta}_i = \frac{1}{n} \sum_{j=1}^n \sigma_j, \quad (2)$$

which is exactly a distributed average consensus, which will converges if the following conditions are met, when

$$\lim_{t \rightarrow \infty} \hat{\beta}_i(t) = \left( \frac{1}{n} \sum_{i=1}^n \sigma_i \right) \mathbf{1}, \quad (3)$$

convergence proof is shown in [29], both the sensor network must be connected and  $\rho(\mathcal{L}_w(\mathcal{G})) < \frac{2}{\Delta}$ , where  $\rho(\mathcal{L}_w(\mathcal{G}))$  correspond to the maximum eigenvalue of the graph in absolute values.

Given the consensus is applied to a multi-sensor system which aim is human victims detection, it is highly required to avoid uncertain measurements produced by different objects in the disaster scene or special situations proper of the emergency area as a fire, smoke, or debris among others. Because of this possible bias measurement and the relevance of human life, we propose the consensus calculation using the best detection measurement acquired until the current moment in any sensor of the network. As depicted in 4,

$$\sigma_i(t) = \max(\sigma_i(k)), \quad (4)$$

where  $k = [0, 1, 2, \dots, t]$ . This consensus calculation will be called M-EC and it will be used to maximize the victim detection level in scenarios with a high uncertainty levels of sensor measurements.

#### 4. SIMULATION AND RESULTS

In order to perform experiments where navigation, sub-swarm generation, consensus formation, and visual victim detection algorithms can be evaluated, it was performed in a virtual scenario with trees, human victims, fire, and quadrotors in uneven terrain. The virtual scenario was developed using a combination of MATLAB, Python, and V-Rep. MATLAB was used to implement the mathematical model for navigation and consensus algorithms. Python performed the CNN model and it was in charge of the human victims' detection. Finally, V-Rep is a virtual robotics environment used to develop a virtual disaster scenario where quadrotor models can be used in SAR operations. The principal reason to perform these experiments in a virtual environment simulation is because of the difficulty to have a real disaster scenario where is possible to perform this kind of experiments, as depicted previously by [2]. Two cases were performed to show the benefits of using the EC and the M-EC. The objective of this simulation is to illustrate the improvement that the use of M-EC implies in cases where the environment generates high levels of uncertainty in the sensors measurement. The first case is a disaster scenario where there are occlusion points generated by objects of the scene, such as debris or trees as shown by Figure 4. In the second case, the same scene is used with the addition of fire and smoke, which increase the sensing uncertainty and it makes it difficult for the victims' detection process as depicted in Figure 5.

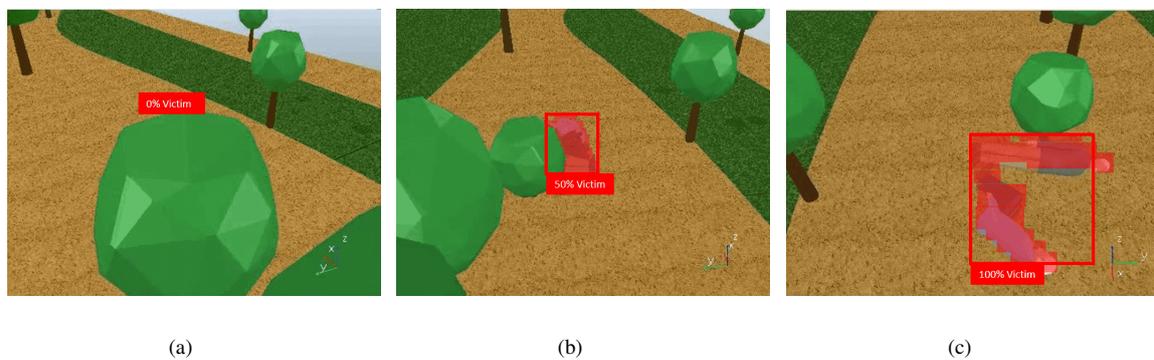


Figure 4. Victim detection, (a) Total occlusion of the victim (0% VDL), (b) Partial occlusion of the victim (50% VDL), and (c) Victim totally detected (100% VDL)

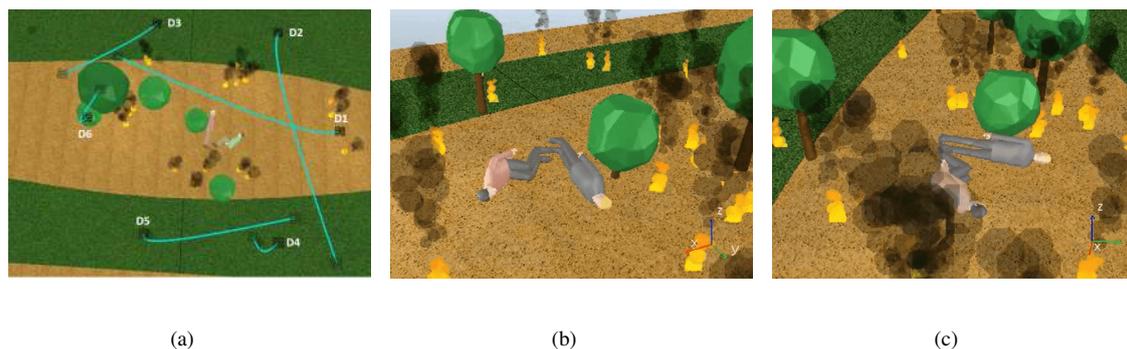


Figure 5. Fire field and victims detection from different drones, (a) The formation consensus, (b) Victim observation by the quadrotor D1, and (c) Victim observation by the quadrotor D2

##### 4.1. Victim detection performed by a single quadrotor

As explained in section 3 every quadrotor is equipped with cameras as a sensing system, which principal aim is human victims detection. The image processing task is performed by a CNN, which is in charge of image analysis focusing on the identification of the potential human victims. As shown in [30] the victims are presented in a urban search and rescue like environments thus the victims will be under rubble or some kind of object will obstruct a percentage of the victim so is important to train a CNN with this kind of data in order to get a better detection, but to get the amount of images that meet this characteristics is a hard task

by itself, in the paper they create a set of 570 images, so it can be consider a small dataset. Transfer learning is a quite useful method for object detection where a feature extractor is use then top layers for classification are fine tune for the task in hand, the need of a dataset that can generalize a desire concept could be difficult to acquire so a virtual environment can be use due to the fact that is more flexible, labels or bounding boxes can be extracted in a automated manner and we can get models of any existing object if needed, in [31] the aim of the paper is to combine this two methods using transfer learning and a virtual dataset to get a pedestrian detector, the results show that high performance can be achieve in real world datasets when the training was done on a virtual dataset then a small set of real images are used for fine tuning. Finally in [32] a CNN is fully train solely with a virtual dataset of three classes then tested in real images, the architecture of this CNN was taken as an initial point for this paper after iterations and pruning we have arrive to the model presented here, where the training time was lower due to the network been shallow without lowering the accuracy.

For the case of this simulation, the topology of CNN consists of 3 convolutional neural layers and two fully connected layers. The convolutional layers have 256 filters each one and  $5 \times 5$ ,  $3 \times 3$ , and  $3 \times 3$  kernel size respectively. The fully connected layers have 512 and 256 neurons respectively with a linear rectifier as the activation function. Finally, the output layer has a neuron in charge of providing the certainty level related to the human victim detection. As depicted in Figure 4 where three cases of victim detection are shown with its respective victim detection level (VDL).

Once the quadrotors are determined as a part of the sub-swarm, all of them perform a formation consensus were every one navigates through the area near to the potential victim and covering the bigger possible area around it as shown by Figure 2. While formation consensus is performed, the VDL changes according to the visibility and proximity of the quadrotor to the potential victim as depicted by Figures 4 and 5. Figure 5a shows the path followed by six quadrotors that conform sub-swarm and Figures 5b and 5c show the victim detection performed by quadrotors D1 and D2 respectively during the formation consensus.

As shown by Figure 6, the measurements provided by different quadrotors about existence of victims in the near area can be confusing and dissimilar. For example, Figure 6a depicts the six different quadrotors sensor cases where D1 is a quadrotor that start its navigation with a total lack of evidence of victim detection, however, while the formation consensus is performed the victim detection improves considerably. D4 depicts a low and constant detection level all the time. Finally, D5 is a quadrotor which loses visual contact with victims but through the time recovers some certain VDL. This is just one example of the wide variety of possible cases that can arise in a real disaster scenario.

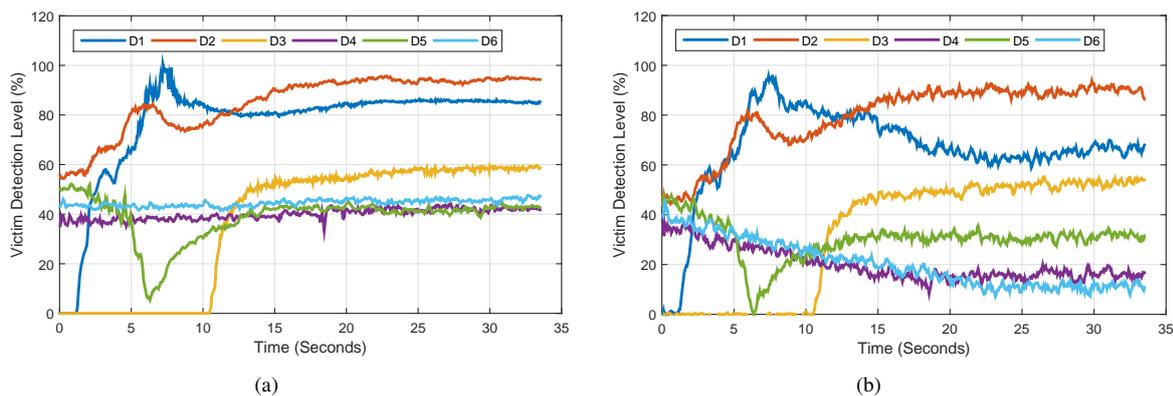


Figure 6. Sensors VDL in a clear and fired scenery, (a) VDL of the six quadrotors in a clear scenery (b) VDL of the six quadrotors in a scenery with fire and smoke

#### 4.2. Victim detection performed by a sub-swarm

As depicted in section 4.1 the single measurement provided by a single quadrotor has a considerable discrepancy if it is compared with measurements provided by the rest of the sub-swarm agents. This discrepancy is just logical if it is taken into account that a disaster place is a chaotic environment where a measurement can be affected by fire, debris, electromagnetic interference, and landslides among others. Figure 6 shows the different VDL values for each quadrotor from sub-swarm during their trajectory in the formation consensus. Figures 6a and 6b show the VDL values according to the two cases described in the

introduction of section 4. Figure 6a shows the VDL of the six quadrotors in a scenario with some occasional occlusions and Figure 6b shows the same scenario but with the addition of fire and smoke, which increases the uncertainty of the VDL and makes it difficult the victims detection. Both cases are evaluated in Table 1, where different statistics descriptors are evaluated from the VDL acquired while the formation consensus event runs. The statistics descriptors used are mean, standard deviation, maximum, and final value. These descriptors show how fire and smoke affect the victim's detection. The VDL average decreases for each quadrotor in the presence of fire and smoke, in a similar way, the standard deviation increases slightly in the second case. This shows that fire reduces the VDL and increases the uncertainty of the measurements. The maximum and final VDL values vary slightly because they also depend on the path that each quadrotor follows in the consensus formation and the presence of fire and smoke in the described path by each quadrotor. This possible lack of agreement between all the sub-swarm quadrotors demonstrates the need for an EC in the way to have a concerted VDL.

Table 1. Detection value by drone

Drone	No-Fire				Fire & Smoke			
	Mean	Std. Dev.	Max	Final	Mean	Std. Dev.	Max	Final
D1	77.05	19.71	99.12	85.26	66.29	22.21	99.62	65.45
D2	85.18	11.42	95.72	94.38	79.94	13.11	92.20	89.41
D3	36.79	25.82	59.66	58.02	32.96	27.29	55.03	54.55
D4	39.98	2.08	44.00	41.72	19.92	8.35	42.09	15.42
D5	38.84	8.80	52.09	42.57	29.74	10.84	48.76	30.25
D6	44.53	1.33	47.61	46.95	20.04	9.02	42.35	11.93

As previously shown, there are many factors that can increase the sensing uncertainty in a disaster area and at the same time make it difficult to detect victims. For this reason, we propose the implementation of an EC to try to agree on an official value of victims detection among the different sub-swarm agents. Additionally, we propose a consensus-based on the maximum values of VDL aimed to offset the effect of adverse factors in the disaster area such as fire, smoke, debris, and collapse among many others. Figure 7 shows the contrast between the EC and the M-EC in both simulation scenarios. Figure 7 shows the EC and M-EC as red and blue lines receptively. Figures 7a and 7b depicts the values of EC and M-EC for both cases performed by disaster zone simulations In Figure 7 is depicted how the M-EC slightly improves the final value of victims detection, going from 61.483% to 66.36%, presenting an increase of approximately 5% which it is not significantly large, however, the second case where there are adverse factors such as fire, the EC decreases considerably, reaching values of 44.5%, as Table 1 previously suggested. In this case, the M-EC considerably improves the estimation, increasing the consensus value up to 63.342% giving and improvement close to 20%, despite of difficulties proper of a disaster zone as fire and smoke.

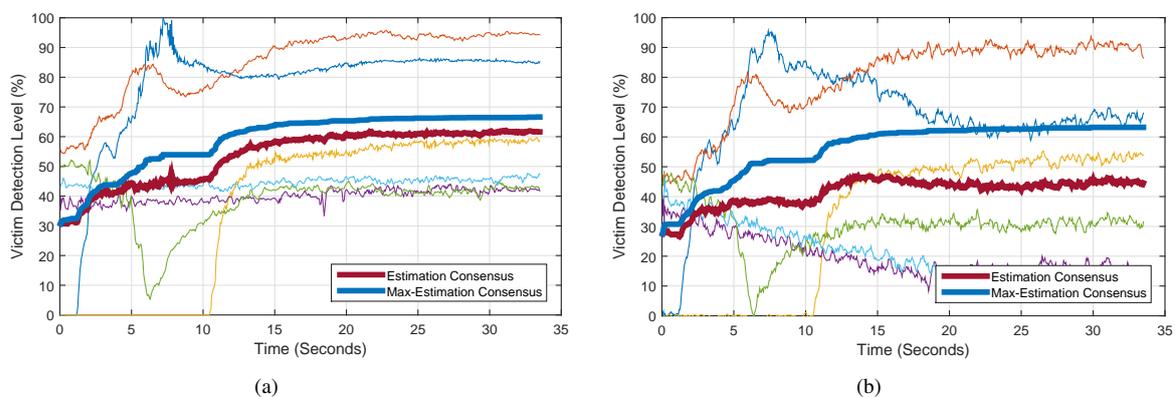


Figure 7. Fire field and victims detection from different drones, (a) EC and M-EC in a clear scenery, (b) EC and M-EC in a scenery with fire and smoke

## 5. CONCLUSIONS AND FUTURE WORK

As expected, the artificial potential functions work well to navigate in a non-convex environment, allowing quadrotors to maintain communication connectivity while avoiding collisions to both other robots and obstacles. The sub-swarm generation makes the swarm to not being stuck by the quadrotors that detect the victim and on the contrary, keeping it navigating. On the other hand, the sub-swarm that detects the victim breaks communication with the big swarm due to they switch the task, from navigating to acquire more information to improve accuracy about the determination of the victim. Every quadrotor was equipped with cameras for victim detection. The visual system includes a CNN in charge of localizing the possible victim and provide an estimation value of the victim called victim detection level (VDL). It was evident that the performance of visual detection can be affected by external aspects and proper of the disaster zone such as fire, smoke, visual occlusions, and debris among others, which can generate mistakes in the VDL generation.

Taking into account that lives are the main objective in SAR missions and mistakes must be diminished, EC and M-EC were introduced, which the main objective is to agree to an estimation from the different measurements provided by each sub-swarm quadrotor. The basic EC indicated to be more effective in detecting victims than the single quadrotor sensing system, however, it showed a reduction of the victim detection consensus level in environments with fire and smoke. Instead, M-EC demonstrates to be robust in environments with visual occlusions, fire, and smoke. Since, if one  $q$  quadrotor fails at identifying a victim, the distributed fashion will assure this mismeasurement will not miss a real victim in the environment. As future work we consider the use of different kinds of sensors that can be in the same estimation network, which makes the graph to be heterogeneous, changing the dynamics of it. Additionally, when different kinds of sensors are used in the estimation, they might have different accuracy levels which can be modeled in the graph as weighted links, indicating which sensors are more reliable than others.

## REFERENCES

- [1] J. Casper and R. R. Murphy, "Human-robot interactions during the robot-assisted urban search and rescue response at the world trade center," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 33, no. 3, pp. 367-385, 2003.
- [2] A. Denker and M. C. 'Is,eri, "Design and implementation of a semi-autonomous mobile search and rescue robot: Salvor," *2017 International Artificial Intelligence and Data Processing Symposium (IDAP)*, pp. 1-6, 2017.
- [3] G. A. Cardona, D. Yanguas-Rojas, M. F. Arevalo-Castiblanco, and E. Mojica-Nava, "Ant-based multi-robot exploration in non-convex space without global-connectivity constraints," *18th European Control Conference (ECC)*, 2019, pp. 2065-2070.
- [4] T. Gunn and J. Anderson, "Dynamic heterogeneous team formation for robotic urban search and rescue," *Journal of Computer and System Sciences*, vol. 81, no. 3, pp. 553-567, 2015.
- [5] J. Le'on, G. A. Cardona, A. Botello, and J. M. Calder'on, "Robot swarms theory applicable to seek and rescue operation," *International Conference on Intelligent Systems Design and Applications*, 2016, pp. 1061-1070.
- [6] T. Takeda, K. Ito, and F. Matsuno, "Path generation algorithm for search and rescue robots based on insect behavior-parameter optimization for a real robot," *IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, pp. 270-271, 2016.
- [7] C. Castillo and C. Chang, "A method to detect victims in search and rescue operations using template matching," *IEEE International Safety, Security and Rescue Robotics, Workshop*, pp. 201-206, 2015.
- [8] Y. Uzun, M. Balcilar, K. Mahmoodi, F. Davletov, M. F. Amasyali, and S. Yavuz, "Usage of hog (histograms of oriented gradients) features for victim detection at disaster areas," *8th International Conference on Electrical and Electronics Engineering (ELECO)*, pp. 535-538, 2013.
- [9] A. Nezirovic, A. G. Yarovoy, and L. P. Lighthart, "Signal processing for improved detection of trapped victims using uwb radar," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 48, no. 4, pp. 2005-2014, 2009.
- [10] Z. Uddin and M. Islam, "Search and rescue system for alive human detection by semi-autonomous mobile rescue robot," *2016 international conference on innovations in science, engineering and technology (ICISSET)*, 2016, pp. 1-5.
- [11] Y. Cui, J. Ren, W. Du, and J. Dai, "Uav target tracking algorithm based on task allocation consensus,"

- Journal of Systems Engineering and Electronics*, vol. 27, no. 6, pp. 1207–1218, 2016.
- [12] Z. Zhou, J. Feng, B. Gu, B. Ai, S. Mumtaz, J. Rodriguez, and M. Guizani, “When mobile crowd sensing meets uav: Energy-efficient task assignment and route planning,” *IEEE Transactions on Communications*, vol. 66, no. 11, pp. 5526–5538, 2018.
- [13] L. G. Jaimes and J. M. Calderon, “An uav-based incentive mechanism for crowdsensing with budget constraints,” *IEEE 17th Annual Consumer Communications and Networking Conference (CCNC)*, 2020, pp. 1–6.
- [14] H. Lv, T. Jiao, Y. Zhang, Q. An, M. Liu, L. Fulai, X. Jing, and J. Wang, “An adaptive-mssa-based algorithm for detection of trapped victims using uwb radar,” *IEEE Geoscience and Remote Sensing Letters*, vol. 12, no. 9, pp. 1808–1812, 2015.
- [15] P. Lorenz and G. Steinbauer, “The robocup rescue victim dataset,” *IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, pp. 1–6, 2018.
- [16] X. Dai, B. Hao, and L. Shao, “Self-organizing neural networks for simultaneous localization and mapping of indoor mobile robots,” *First International Conference on Intelligent Networks and Intelligent Systems*, pp. 115–118.
- [17] G. De Cubber and G. Marton, “Human victim detection,” *Third International Workshop on Robotics for risky interventions and Environmental Surveillance-Maintenance, RISE*, 2009.
- [18] Y. S. Dadwhal, S. Kumar, and H. Sardana, “Data-driven skin detection in cluttered search and rescue environments,” *IEEE Sensors Journal*, vol. 20, no. 7, pp. 3697–3708, 2019.
- [19] S. Lee, D. Har, and D. Kum, “Drone-assisted disaster management: Finding victims via infrared camera and lidar sensor fusion,” *3rd Asia-Pacific World Congress on Computer Science and Engineering (APWC on CSE)*, pp. 84–89, 2016.
- [20] C. L. Giles and K.-C. Jim, “Learning communication for multi-agent systems,” *Workshop on Radical Agent Concepts. Springer*, pp. 377-390, 2002.
- [21] T. Xu, N. A. Shevchenko, D. Lavery, D. Semrau, G. Liga, A. Alvarado, R. I. Killely, and P. Bayvel, “Modulation format dependence of digital nonlinearity compensation performance in optical fibre communication systems,” *Optics Express*, vol. 25, no. 4, pp. 3311–3326, 2017.
- [22] W. O. Quesada, J. I. Rodriguez, J. C. Murillo, G. A. Cardona, D. Yanguas-Rojas, L. G. Jaimes, and J. M. Calderon, “Leader-follower formation for uav robot swarm based on fuzzy logic theory,” *International Conference on Artificial Intelligence and Soft Computing*, 2018, pp. 740–751.
- [23] G. A. Cardona and J. M. Calderon, “Robot swarm navigation and victim detection using rendezvous consensus in search and rescue operations,” *Applied Sciences*, vol. 9, no. 8, 2019.
- [24] G. Cardona, D. Tellez-Castro, and E. Mojica-Nava, “Cooperative transportation of a cable-suspended load by multiple quadrotors,” *IFAC-PapersOnLine*, vol. 52, no. 20, pp. 145–150, 2019.
- [25] D. Mellinger and V. Kumar, “Minimum snap trajectory generation and control for quadrotors,” *IEEE international conference on robotics and automation*, 2011, pp. 2520–2525.
- [26] R. Mahony, V. Kumar, and P. Corke, “Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor,” *IEEE robotics and automation magazine*, vol. 19, no. 3, pp. 20-32, 2012.
- [27] T. Lee, M. Leok, and N. H. McClamroch, “Geometric tracking control of a quadrotor uav on se (3),” *49th IEEE conference on decision and control (CDC)*, 2010, pp. 5420-5425.
- [28] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” *Proceedings of the IEEE*, 1998, vol. 86, no. 11, pp. 2278–2324.
- [29] M. Mesbahi and M. Egerstedt, “Graph theoretic methods in multiagent networks,” *Princeton University Press*, 2010.
- [30] A. Fung, L. Y. Wang, K. Zhang, G. Nejat, and B. Benhabib, “Using deep learning to find victims in unknown cluttered urban search and rescue environments,” *Current Robotics Reports*, pp. 1–11, 2020.
- [31] L. Ciampi, N. Messina, F. Falchi, C. Gennaro, and G. Amato, “Virtual to real adaptation of pedestrian detectors for smart cities,” *arXiv preprint arXiv:2001.03032*, 2020.
- [32] E. Bochinski, V. Eiselein, and T. Sikora, “Training a convolutional neural network for multi-class object detection using solely virtual world data,” *13th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, 2016, pp. 278–285.