Distributed Target Identification in Robotic Swarms

Paolo Stegagno
Max Planck Institute for biological Cybernetics
Spemannstr. 38
Tübingen, Germany
pstegagno@tuebingen.mpg.de

Caterina Massidda
Max Planck Institute for biological Cybernetics
Spemannstr. 38
Tübingen, Germany
cmassidda@tuebingen.mpg.de

Heinrich H. Bülthoff
Max Planck Institute for biological Cybernetics
Spemannstr. 38
Tübingen, Germany
hhb@tuebingen.mpg.de

ABSTRACT

The ability to identify the target of a common action is fundamental for the development of a multi-robot team able to interact with the environment. In most existing systems, the identification is carried on individually, based on either color coding, shape identification or complex vision systems. Those methods usually assume a broad point of view over the objects, which are observed in their entirety. This assumption is sometimes difficult to fulfill in practice, and in particular in swarm systems, constituted by a multitude of small robots with limited sensing and computational capabilities. In this paper, we propose a method for target identification with a heterogeneous swarm of low-informative spatially-distributed sensors employing a distributed version of the naive Bayes classifier. Despite limited individual sensing capabilities, the recursive application of the Bayes law allows the identification if the robots cooperate sharing the information that they are able to gather from their limited points of view. Simulation results show the effectiveness of this approach highlighting some properties of the developed algorithm.

1. INTRODUCTION

Robotic swarms and multi-robot systems are receiving growing interest from the scientific community due to their intrinsic easiness of design and manufacture, versatility, robustness to hardware failures, and capability to execute spatially distributed tasks. Understanding and controlling a swarm of this kind poses a range of research challenges that are still to be addressed for robust real world applications that demand several coordinated components involving control, localization, sensing and interpretation of the surrounding environment.

Among other issues in multi-agent perception, target identification is still one of the less studied, but most important in order to let the system interact with the environment (e.g.: to individuate the object of a common action). Several peculiarities of robotic swarms must be taken into account. First, the physical dimension of platforms considered in swarm robotics is usually one or two orders of magnitude smaller than the typical dimension of the objects in the environment. In addition, constraints on payload, power consumption and computational power limit the exteroceptive sensor equipment of the robots. Hence, each robot is usually able to observe only some details (e.g.: one or two edges) or some specific feature (e.g.: color or material) of the sensed objects. At the current size of typical swarm platforms (few to tenths of centimeters), this problem arises when the robots are required to identify for example cars, trees or buildings. However, considering the current miniaturization trend, soon the same problem will consider smaller objects of the environment as office furniture or tools of everyday life.

Being identification usually considered a computer vision topic, most of the state-of-the-art algorithms (e.g.: [16], [12], [2]) make use of highly-informative sensors as cameras or 3D range finders. Many robotic systems integrating the identification of an object (e.g.: [18], [4]) assume a broad point of view (hence one single sensor is enough for the recognition), perform the computations in a centralized fashion and are computationally expensive and real-time-unfeasible on limited platforms. The idea of using local features of the objects in order to perform their identification has been largely studied in literature [12], [2], [15], [9], but it has been exploited mostly by means of a centralized entity and considering only visual features.

A generalization of this approach is multiple-view scene interpretation, which studies the problem of understanding the environment given the images collected from different points of view. The algorithms developed to address this problem (e.g.: [17]) usually take advantage of common features from the different views. In [1] the authors propose a method for the selection of an optimal number of images taken from different views of a 3D object in order to perform the recognition. The authors of [19] propose two algorithms for object recognition based on matching of scale invariant features. In [6] the authors exploit the multi-view aspect of the problem in order to reconstruct the cluttered parts of the environment and discern the subject from the background. In [14], the authors propose a recognition method to jointly classify the object observed by a network of smart cameras. It is important to underline that most methods are able to select informative local features of the observed objects, thanks to their broad point of view, whereas each small agent of a swarm can only rely on its limited, non-selected point of view.

SAC'15, April 13-17, 2015, Salamanca, Spain

http://dx.doi.org/10.1145/2695664.2695922

In multi-robot systems, many authors ([5], [7]) have dealt with the problem of jointly estimating the location of a common target by fusing the individual readings of the robots. However, each robot still enjoys a wide point of view over the scene. In [13], the authors developed a system to classify objects based on sound features and visual informations. For the best of our knowledge, this paper is the first to explicitly address the problem of target identification in robotic swarms even if several peculiarities of this type of systems calls for ad-hoc algorithms. Its role is to introduce and formalize the problem of target identification by means of multiple heterogeneous low-informative and spatially distributed sensors, also proposing a distributed system for its solution based on probability theory.

The rest of the paper is organized as follows. Section 2 formally introduces the problem of target identification by means of multiple low-informative spatially-distributed sensors. The developed method is presented in Sections 3, 4 and 5, which describes respectively the classification algorithm, the type of measurements and the communication and distribution of the computation among the team members. Section 6 presents the first 2D implementation in simulation, and Section 7 concludes the paper.

2. PROBLEM FORMULATION

Consider a heterogeneous swarm system A of n agents $A = \{\mathcal{A}_1, \mathcal{A}_2, \ldots, \mathcal{A}_n\}$ living in a generic environment and surrounding an object ω in such environment. Robots are not aware of each other's position, nor of their absolute positions in the world. We do not consider the problem of localization, since our method will not use relative or absolute positions. Similarly, in this paper we do not study the problem of controlling the swarm so as to reach a situation as described by our assumptions. In general, we assume that the robots are able to move in swarm and to surround an object avoiding collisions and maintaining communication connectivity.

Each $\mathcal{A}_i$ is equipped with an exteroceptive sensor and gathers a measurement z_i of ω. In general, different robots can be equipped with different sensors, and z_i should be considered general. Possible types of sensors include cameras, laser range finders, sonar and IR arrays, material detectors, temperature and stiffness sensors among others. We will discuss three specific types of sensors in Section 4, whose simulation will be used to numerically validate the proposed algorithms. In Section 4, we will take some additional assumptions on the position assumed by each robot on the basis of the different types of sensors. We denote by $Z = \{z_i, i = 1, \ldots, n\}$ the set of measurements collected by all robots.

Moreover, we assume that robots are able to communicate with all other robots within a given communication ray C_r. As a consequence, the communication graph is undirected, i.e.: if a $\mathcal{A}_i$ communicates with $\mathcal{A}_j$, then also $\mathcal{A}_j$ communicates with $\mathcal{A}_i$. The communication graph is defined as a pair of sets of nodes $\mathcal{N}$ and edges $\mathcal{E}$. Each $\mathcal{A}_i$ is represented as a node $i \in \mathcal{N}$, while an edge $(i, j) \in \mathcal{E}$ means that $\mathcal{A}_i$ is communicating with $\mathcal{A}_j$ and vice versa. We assume the communication graph to be connected.

We assume that in the world in which the robots live there is one and only one object, denoted by ω, out of a set of m possible types of objects $\Omega = \{\omega_1, \omega_2, \ldots, \omega_m\}$. Each ω_j is identified by a label $l_j = j$, with $L = \{l_1, l_2, \ldots, l_m\}$. Finally, we assume that the measurement z_i gathered by each generic $\mathcal{A}_i$ refers to ω, hence in general Z is implicitly $Z(\omega)$ a function of ω. We can now define the identification problem as follows.

PROBLEM 1. *The problem of identifying the object ω out of the set Ω through the measurements $Z = \{z_i, i = 1, \ldots, n\}$ is the problem of assigning to ω a label $l = \mathrm{cl}(Z)$ out of the set L on the basis of the measurements Z, where $\mathrm{cl}(Z) = l_j$ if and only if ω is recognized to be of type ω_j.*

Let be $O(\omega)$ a random variable representing the type of the object:

$$O(\omega) = O = j \iff \omega = \omega_j \tag{1}$$

Then we can define the probability $p(O = j)$ as the probability that ω is of type ω_j. A common policy to solve Problem 1 and assign a label to ω is to take the final decision on the classification as the value which maximizes it:

$$\mathrm{cl}(Z) = \underset{l \in L}{\operatorname{argmax}}\, p(O|Z). \tag{2}$$

Whenever a labeling policy $\mathrm{cl}(Z)$ is given in this form, it is referred to as *Bayes classifier*. The system must estimate $p(O|Z)$ in order to take a decision. Then, the focus of this work will be the distributed estimation of $p(O|Z)$.

3. NAIVE BAYES CLASSIFICATION

Even if the classification rule (2) seems simple, a characterization of the conditional probability $p(O|Z)$ is not straightforward. From now on, we will focus on computing an estimate $\hat{p}(O|Z)$ of such probability. Through the application of the Bayes rule, $p(O|Z)$ can be rewritten as:

$$p(O|Z) = \frac{p(O)p(Z|O)}{p(Z)}. \tag{3}$$

The recursive application of the definition of conditional probability leads to the factorization of the numerator of the right side of equation (3) as:

$$\begin{aligned} p(O)p(Z|O) = p(O)p(z_i, i = 1, \ldots, n|O) = \\ = p(O)p(z_1|O)p(z_2|O, z_1) \ldots p(z_n|O, z_1, \ldots, z_{n-1}). \end{aligned} \tag{4}$$

Even if equation (4) allows the recursive computation of $p(O|Z)$ using the measurements one by one, the characterization of the dependency between the z_i's can still prevent the actual computation of each factor. In addition, we aim at designing a distributed scheme. Indeed, those two requirements can be met considering a simplified version of the above algorithm.

In the *naive* Bayes classifier, the measurements are assumed to be conditionally independent from each other. Although such assumption is quite strong, experimental studies have shown the validity of this approach in several problems (e.g.: [11]). Hence, exploiting the conditional independence of the z_i's (i.e.: $p(z_i|z_j) = p(z_i)\, \forall i \neq j$) we can drastically simplify equation (4):

$$p(O)p(Z|O) = p(O) \prod_{i=1}^{n} p(z_i|O). \tag{5}$$

In order to have a complete form for the estimate of $p(O|Z)$, we also note that the denominator of equation (3) is independent from O, hence it is a constant that can be computed

normalizing $p(O|Z)$ such that

$$\sum_{\omega \in \Omega} p(O|Z) = 1. \quad (6)$$

Therefore, by this consideration and by using equation (5), equation (3) can be rewritten as:

$$p(O|Z) = \alpha\, p(O) \prod_{i=1}^{n} p(z_i|O) \quad (7)$$

where α is a factor ensuring the validity of equation (6).

4. MEASUREMENTS

Most of the difficulties in the application of equation (7) are associated with modeling the probability distributions $p(z_i|O)$. In particular, for each type of sensor used to gather information on ω it is necessary to

i) build models of the ω_j's which encase the features of the objects that are measurable by means of that sensor;

ii) exploit these models in order to compute estimates of $p(z_i|O)$.

Note that $p(z_i|O) = p(z_i|O = j)$ expresses the probability of obtaining the measurement z_i given that the object ω is of type ω_j. Then, it is worth to underline that $p(z_i|O)$ can be computed independently for each possible $\omega_j \in \Omega$. In the following, we consider three types of sensors both for simulation and example purposes: extremely low-resolution cameras, laser scanners and relative reflectance sensors.

4.1 Color measurements

Using images of an object in order to recognize it is a well known and studied problem in computer vision. State-of-the-art computer vision techniques for object recognition make use of feature extraction algorithms to select useful information, such as edges, shapes and local features, about the observed objects from the images gathered by cameras.

However, an image at decent resolution (e.g.: 640×480) provides way more information with respect to what a robot with limited computational power is able to handle. In addition, the aforementioned feature extraction techniques usually constitutes an increase of the computational load for the robots, and should be avoided in our setting. Nevertheless, a camera can still provide useful information, such as the color of the sensed object for a relatively low price. For these reasons, we will consider a very low resolution RGB camera with 8×6 pixels only, which will be used in order to sense the color of a limited region of ω. Ideally, the developed method can be applied also with a 1 pixel camera.

If $\mathcal{A}_i$ is equipped with such camera, then z_i is constituted by an ordered set of 8×6 pixels $z_i = \{x_{kl} : k = 1, \ldots, 8, l = 1, \ldots, 6\}$, each pixel $x_{kl} = (r_{kl}\ g_{kl}\ b_{kl}) \in \mathcal{X}^3$, with $\mathcal{X} = [0, 255]$, containing the red, green and blue (RGB) values of the color of a point of the surface of ω. The measured color is affected by measurement error that we assume to be zero-mean and known-covariance Gaussian additive noise. In general, environment illumination condition also affects the colors sensed by cameras, but in our case we assume that the robots are sensing a very small portion of ω from a close distance, so that they are able to illuminate it, drastically reducing such effects which is then considered to be part of the measurement noise.

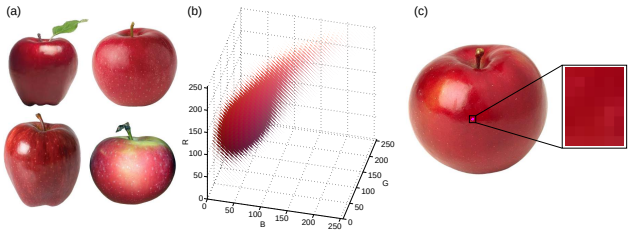

Figure 1: (a) Four objects of type apple; (b) generalized color histogram of the object apple computed using the four images in (a); (c) a test object and the measurement gathered by a robot.

The models of the generic $\omega_j \in \Omega$ used by the robots equipped with cameras is a *normalized color histogram* $h(\omega_i)$ *of* ω_i built on one or more pictures of one or more objects of type ω_j. The *color histogram* $\hat{h}(\mathcal{I})$ of an image $\mathcal{I}$ is a $q \times q \times q$ 3-dimensional matrix, each cell $\hat{h}_{ijk}$ representing the number of pixels of $\mathcal{I}$ whose RGB values are in the set $H_{ijk} = [(i-1)a, ia[\times[(j-1)a, ja[\times[(k-1)a, ka[$, where $a = 256/q$ is the discretization step. The histogram provides a compact summarization of the distribution of data in an image, and has been largely used in computer vision, both for colors [16] and other features [15] [9].

Let be $\bar{h} = \sum_{(i,j,k)} \hat{h}_{ijk}$, then $h(\mathcal{I}) = \hat{h}(\mathcal{I})/\bar{h}$ is the *normalized color histogram* such that $\sum_{(i,j,k)} h_{ijk} = 1$, and each cell h_{ijk} of $h(\mathcal{I})$ contains the probability that picking a random pixel $x = (r\ g\ b)$ of $\mathcal{I}$, then $x \in H_{ijk}$. Note that given a certain $x = (r\ g\ b)$, then the indexes of the cell containing its probability can be easily computed as $i = \lfloor r/a \rfloor, j = \lfloor g/a \rfloor, k = \lfloor b/a \rfloor$, where $\lfloor \square \rfloor$ indicates the floor of the quantity $\square$. Considering the noise on the measurements of the camera, and possible shades of the observed object, then a good practice in order to obtain a more realistic model is to apply a three-directional Gaussian filter $g(\cdot)$ to $h(\mathcal{I})$ in order to have a more smooth normalized histogram $g(h(\mathcal{I}))$.

Considering c different images $\mathcal{I}_1, \ldots, \mathcal{I}_c$ of one or more objects of type ω_j, then the normalized color histogram $h(\omega_j)$ of ω_j can be defined as

$$h(\omega_j) = \frac{1}{c} \sum_{i=1}^{c} g(h(\mathcal{I})). \quad (8)$$

Once $h(\omega_j)$ is known, it is straightforward to compute an estimate of $p(z_i|O = j)$ as the mean of the probability of each pixel in the measurement image z_i:

$$\hat{p}(z_i|O = j) = \frac{1}{8 \cdot 6} \sum_{x \in z_i} h_{\lfloor r/a \rfloor \lfloor g/a \rfloor \lfloor b/a \rfloor}. \quad (9)$$

Note that another feasible estimate $\hat{p}(z_i|O = j)$ could be computed as the product of all $h_{\lfloor r/a \rfloor \lfloor g/a \rfloor \lfloor b/a \rfloor}$, but this would imply the independence of the measurements of the single pixels, which is not a realistic assumption since the measured points of the surface of ω are all close to each other. An example of generalized histogram matrix of an apple is depicted in Fig. 1, as well as the measurement gathered by a robot on a test object.

4.2 Laser scanner measurements

A second feature that we want to exploit in order to iden-

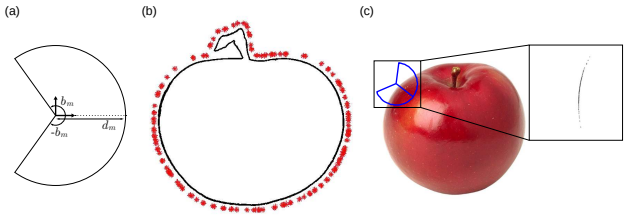

Figure 2: (a) Field of view of a laser scanner; (b) particles selected for the fourth image of Fig. 1a; (c) a test object and the measurement gathered by a robot.

tify ω is its profile. It is undoubtable that the profile of part of an object can provide useful information for its identification. For example, the profile of a box contains corners and straight lines, while the profile of a bottle is a circumference. Performing identification using these features can be done at relatively low computational cost while obtaining interesting results. Then, we chose a laser range finder as second type of sensor, also due to its popularity among the robotics community.

If $\mathcal{A}_i$ is equipped with a laser range finder, then z_i is constituted by an ordered set of bearing–distance pairs $z_i = \{(b\ d) : -b_m \leq b \leq b_m, d \leq d_m\}$ representing the contour of the object in the field of view of the sensor, delimited by $-b_m$, b_m and d_m and depicted in Fig. 2a.

The model of the generic $\omega_j \in \Omega$ used by the robots equipped with laser scanners is a set of particles generated on c images $\mathcal{I}_1, \ldots, \mathcal{I}_c$ of objects of type ω_j. We assume that the generic robot $\mathcal{A}_i$ equipped with a laser scanner is able to position itself with a limited error at a fixed distance d_f from ω. This detail largely reduces the number of needed particles in order to model each ω_j, speeding up both the offline modeling of ω_j and more importantly the online computation of $p(z_i|O = j)$. In fact, in order to build the model of ω_j, we extract n_c random particles for each $\mathcal{I}_k, k = 1, \ldots, c$, at distance d_f from the contour of the object and oriented in the direction of the center of the object. In Fig. 2b we show the particles generated for the fourth image of Fig. 1a.

Then, considering c images, the model of ω_j is a set $P_t = \{p_t, t = 1, \ldots, cn_c\}$ of cn_c particles, each particle p_t containing the measurement z_i that $\mathcal{A}_i$ would gather if it was in the corresponding configuration. By the knowledge of the model of the sensor (i.e.: the noise on the measurements and the noise on the positioning of the robot) it is easy to compute $p(z_i|p_t)$. Then, the estimate of $p(z_i|O = j)$ is:

$$\hat{p}(z_i|O = j) = \max_{p_t \in P_t} p(z_i|p_t) \tag{10}$$

In this case, we chose to select the maximum among the particles in order to magnify the effect of peculiar parts of the objects. Another viable solution would have been to compute the average over all particles of an object. In both cases, equation (10) will produce higher probabilities for the objects whose models include particles comparable to the collected measurement. An example of measurement from the laser scanner is reported in Fig. 2c.

4.3 Reflectance measurements

Another feature that can be used to discriminate among objects is their material. Even though there is no sensor able to measure directly a material, there are methods to measure useful quantities in order to infer the material of a sensed surface. State-of-the-art approaches range from analysis of contact vibrations [11] to analysis of acoustic features [20].

A promising field derived by remote sensing uses multi- and hyper-spectral analysis (e.g.: [10]). As a particular material usually has a specific color given by the different reflection rate of electromagnetic (EM) radiation depending on the wavelength in the visible spectrum (390-700 nm), this behavior holds also in the non-visible part of the EM spectrum. Hence, considering the whole spectrum in order to discriminate objects may give a strong advantage with respect to considering only the colors derived by the visible light. However, precise multi- and hyper-spectral equipments are usually heavy and extremely expensive. Moreover, the huge amount of information that this type of camera can provide prevents its application in robotic swarms.

Nevertheless, there exist relatively cheap sensors (e.g.: ALTA II reflectance spectrometer) that can measure the relative reflectance (the ratio of the reflected EM radiation over the incident EM radiation, from now on only reflectance) of a small portion of an object at one or more specific wavelengths. The working principle is simple: enclose a small portion of an object with a probe so that there is no external EM radiation impacting on that surface, emit a radiation at a given wavelength with a suitable LED and measure the percentage of radiation that is reflected. Given its simplicity, this sensor is suitable for the application in robotic swarms and we will consider it as third type of sensor.

Let be $\mathcal{A}_i$ a robot equipped with a reflectance sensor, then $z_i = r$, $r \in \mathbb{R}_0^+$ is a single value expressing the reflectance of ω in the position of $\mathcal{A}_i$ for a wavelength of 810 nm, where $r = 0$ means no reflection at all and $r = 1$ means complete reflection. Note that the reflectance can also be $r > 1$, meaning that the object reflects more EM radiation than the incident one (i.e.: it emits radiation at the selected wavelength). We assume that r is affected by zero-mean known-covariance Gaussian noise. We chose the wavelength of 810 nm because it is outside of the visible spectrum, hence it provides information not already contained in the colors, but it is in the Near-Infra-Red (NIR) spectrum, close to visible radiation, hence it is easy to measure with cheap sensors.

In addition, the values of reflectance in visible and NIR spectra of many objects and materials are available in scientific literature (e.g.:[8], [3]) and on the web. Then, it is possible to create realistic models of several objects even without using the actual sensor. The first step consists in preparing grayscale images of objects of type ω_j containing in each pixel a value representing the reflectance of those objects at the chosen wavelength in the corresponding point of the objects. Since the relative reflectance values of most objects are in the range between 0 and 1, and the values of the pixels of a grayscale image range from 0 to 255, a natural choice is to scale the reflectance values by a constant $k = 200$, such that a pixel value of 0 correspond to a reflectance of 0% and a pixel value of 200 correspond to a reflectance of 100%. This choice gives also the possibility to include values greater than 100% in order to model objects that emit EM radiation at the selected wavelength. Examples of such images are reported in Fig. 3a for the objects depicted in Fig. 1a.

Through those images $\mathcal{I}_1^r, \ldots, \mathcal{I}_c^r$, it is possible to build a *normalized reflectance histogram* $h^r(\omega_j)$ *of* ω_j with the same method described for the normalized color histogram. First

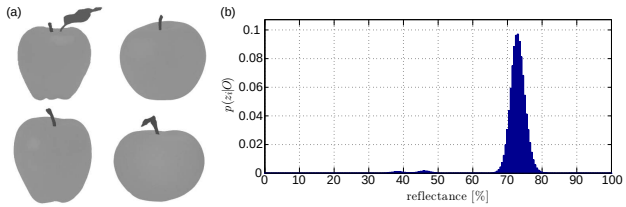

Figure 3: (a) Grayscale images used to build the model of the apple; (b) resulting generalized reflectance histogram.

we compute the *reflectance histogram* $\hat{h}^c(\mathcal{I}_l^r)$ of each $\mathcal{I}_l^r$ as a q-dimensional vector, each cell $\hat{h}_i^c$ representing the number of pixels of $\mathcal{I}_l^r$ whose grayscale values are equal to i. Then, the normalized reflectance histogram $h^r(\omega_j)$ of ω_j is

$$h^r(\omega_j) = \frac{1}{c}\sum_{l=1}^{c} g(h(\mathcal{I}_l^r)). \tag{11}$$

Once $h^r(\omega_j)$ is known, it is straightforward to compute an estimate of $p(z_i = r|O = j)$ as:

$$\hat{p}(z_i|O = j) = h^c_{(200r)}. \tag{12}$$

The generalized reflectance histogram computed using the four images in Fig. 3a is reported in Fig. 3b.

5. DISTRIBUTED IMPLEMENTATION AND COMMUNICATION

One of the features of swarm systems is the possibility to distribute the computational load on multiple robots, although the way this distribution can be implemented is not always straightforward. Luckily, the recursive formulation of the naive Bayes classifier (equation (7)) offers an easy paradigm for distributing the computation among the components of the team. Each $\mathcal{A}_i$ can compute the probability distribution $p(z_i|O)$ associated to its measurement and then communicate it to its mates. This information, thanks to the factorization expressed by equation (7), is all that is needed to compute the probability distribution of the objects, hence to classify ω among the set of the known objects.

This approach, despite its simplicity, brings several advantages. First, the computational load of computing all the $p(z_i|O)$ is distributed among the robots. Second, the messages exchanged by the robots are all of the same type, containing $p(z_i|O)$ and not directly the measurements. Last, but not less important, each measurement z_i remains local on $\mathcal{A}_i$, so that $\mathcal{A}_i$ has to deal only with one type of measurements: the one it gathers. This means that the robots equipped with a certain type of sensor need only to know the models of the $\omega_j \in \Omega$ suitable for their own sensor. The measurements coming from the other robots, being already transformed in probabilities, does not require any previous knowledge (i.e.: model). Then, for example, it is also possible to add at run-time new sensors without modifying the already set up robots.

The drawback of this choice is the necessity to implement a suitable communication scheme in order to spread the information in the team. At this aim, a classical gossiping algorithm can be used. For simplicity, we assume that the communication among robots happens at fixed time instants $t = T, 2T, \ldots, kT, \ldots$ with T communication period, and we assume that all the robots broadcast the information to their communication neighbors simultaneously. The information sent from $\mathcal{A}_i$ to $\mathcal{A}_j$ at the communication step k (i.e.: $t = kT$) will be available to $\mathcal{A}_j$ at step $k+1$. Then, the gossiping algorithm can be implemented following two simple rules:

i) at the first communication step, each $\mathcal{A}_i$ sends its $P(z_i|O)$ to the neighbors;

ii) if in the k-th communication step $\mathcal{A}_i$ receives $P(z_j|O)$ for the first time, at the $(k+1)$-th communication step $\mathcal{A}_i$ communicates $P(z_j|O)$ to its neighbors.

This algorithm can be also generalized to the case of non simultaneous communication to take into account a more realistic model of communication and possible dropped messages. However, the optimization of the communication scheme goes beyond the scope of this paper.

Denote by A_k^i the set of robots whose probabilities $p(z_j|O)$ are known to $\mathcal{A}_i$ at step k, and let be $Z_k^i = \{z_j : \mathcal{A}_j \in A_k^i\}$ the set of their measurements. At each step k, $\mathcal{A}_i$ can compute $p(O|Z_k^i)$ through equation (7), until $A_k^i = A$ and then $p(O|Z) = p(O|Z_k^i)$. However, a more efficient way to compute $p(O|Z)$ is to iteratively update $p(O|Z_k^i)$ without recomputing every time all the products. For example, assume that $A_{k+1}^i = \{A_k^i, \mathcal{A}_h\}$, which means that at step $k+1$ $\mathcal{A}_h$ receives for the first time $p(z_h|O)$. Then $p(O|Z)$ can be computed through the iterative application of:

$$p(O|Z_{k+1}^i) = \alpha\, p(O) \prod_{j \in A_{k+1}^i} p(z_j|O) =$$

$$= \alpha\, p(O) p(z_h|O) \prod_{j \in A_k^i} p(z_j|O) = \alpha_1 p(O|Z_k^i) p(z_h|O). \tag{13}$$

6. SIMULATIONS

To test the developed system, we have set up a 2D simulation framework and a database of 12 types of objects ω_j (leaf, banana, sunflower, apple, starfish, butterfly, grape, hammer, pineapple, strawberry, wrench, scissors). For each ω_j, four images were used in order to build the models for the three sensors as described in Section 4, hence the whole database is built through 48 images, shown in Fig. 4. We have simulated the measurements as described in Section 4, considering a 8×6 camera, a laser scanner with 200 rays and a field of view limited by $b_m = 120°$, $d_m = 200$. Moreover, we have fixed the discretization step a of the normalized color histogram to $a = 8$ (hence $q = 32$), and we have selected 600 particles (150 per image) to build the models for the laser scanner measurements.

In a single simulation, an object (image) ω is placed in the scene, and multiple robots are randomly deployed over it (image and reflectance measurements) or in its proximity (laser scanner). An example of such simulation is shown in Fig. 6, where the blue shapes are the fields of view of the robots equipped with the laser scanners, the green dots are robots equipped with the reflectance sensors and the magenta dots are the robots equipped with the cameras.

Once the deployment is done, the communication radius is selected as the minimum C_r such that the communication graph is connected. In general, C_r is not a parameter that can be freely chosen, and should be fixed in the beginning of the simulation. However, we chose to minimize it to have as least connections as possible, in order to stress in each simulation the communication among the robots. In Fig. 6

Figure 4: Images used to build the models of the objects (training set).

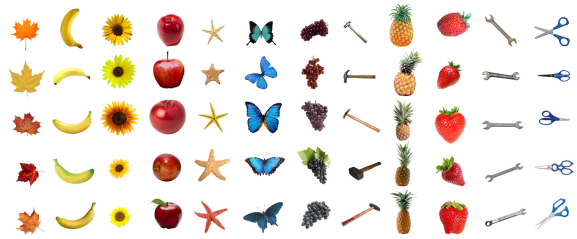

Figure 5: Images used for testing purpose (testing set).

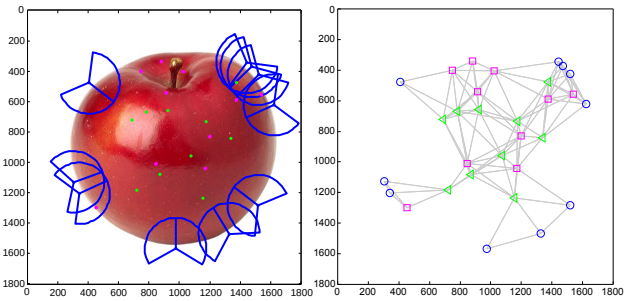

Figure 6: Example of deployment of the robots and the communication graph.

we show also the communication graph corresponding to the presented simulation.

At the beginning of the simulation, each robot collects a measurement and uses the appropriate model set (i.e.: depending on its type of sensor) to compute $p(z_i|O)$. Then, the communication and the iterative application of equation (13) is performed for each robot, until the communication among the robots ends because all robots have received all the information from the other robots and have computed the same values for $p(O|Z)$. In the final step, the robots assign a label to ω based on equation (2).

We have conducted extensive simulations in order to assess the quality of the recognition on a set of 60 images, 5 for each ω_j, shown in Fig. 5 and to test different configurations of the swarm system varying the total number of robots and the type of sensors equipped. In particular, we have tested the system with 6, 15 and 30 robots. For each of those values, we have tested the cases of all robots equipped with cameras, all robots equipped with laser scanner, all robots equipped with reflectance sensors, half of the robots equipped with cameras and half equipped with laser scanner, and finally one third of the robots equipped with each type of sensor, for a total number of 15 different configurations..

In order to have a massive quantity of data, for each of the aforementioned configurations we have performed 20 complete simulations for each image in the testing set (then 100 simulations for each ω_j, and 1200 simulations to test each configuration of the swarm). At the end of each simulation, the type of object selected by the robots for ω is the one that receives the highest probability.

	Le	Ba	Su	Ap	St	Bu	Gr	Ha	Pi	Sr	Wr	Sc
Le	**77**	0	0	0	0	0	1	0	0	0	0	0
Ba	20	**100**	0	0	1	0	0	0	0	0	0	0
Su	1	0	**100**	0	0	0	0	0	0	0	0	0
Ap	0	0	0	**75**	0	0	0	0	0	4	0	0
St	0	0	0	0	**99**	0	0	0	0	0	0	0
Bu	0	0	0	0	0	**100**	0	0	0	0	0	0
Gr	0	0	0	0	0	0	**90**	0	0	0	0	0
Ha	0	0	0	0	0	0	1	**97**	0	0	0	0
Pi	0	0	0	0	0	0	0	0	**100**	0	0	0
Sr	2	0	0	25	0	0	8	0	0	**96**	0	0
Wr	0	0	0	0	0	0	0	3	0	0	**99**	0
Sc	0	0	0	0	0	0	0	0	0	0	1	**100**

Table 1: Confusion matrix for simulations with 30 robots, 10 equipped with cameras, 10 with laser scanners and 10 with reflectance sensors. Abbreviations: Le - Leaf; Ba - Banana; Su - Sunflower; Ap - Apple; St - Starfish, Bu - Butterfly; Gr - Grape; Ha - Hammer; Pi - Pineapple; Sr - Straweberry; Wr - Wrench; Sc - Scissors.

	C	**L**	**R**	**C+L**	**C+L+R**
1 robot	29.5	49.5	53.2	-	-
6 robots	67.5	48.5	80.2	70.3	86.33
15 robots	69.7	56.9	86.3	78	93.6
30 robots	69.1	64.1	90.3	78	94.4

Table 2: Percentage of correct associations for all 15 configurations of the swarm. C+L is camera and laser scanner, C+L+R is camera, laser scanner and reflectance sensor.

The results of the simulations for each configuration can be summarized in a confusion matrix as the ones presented in Table 1, in which the columns and the rows express the actual and estimated types of ω respectively. Each cell contains the percentage of correct identifications for the corresponding case. It is evident that the higher are the numbers on the main diagonal, the better the algorithm performs, while the numbers off the main diagonal suggest similarities among the object that leads to non-correct identifications. This is the case for example of the apple and the strawberry: they are both red, with almost round contours and made of similar organic material (hence producing similar measurements with the reflectance sensor).

A more compact representation of the overall performance of a configuration can be computed as the overall percentage of correct identifications. Those values are reported for all 15 studied configurations in Table 2, and show how in general the performance increases with the number of robots. More interesting, whenever different types of measurements are used together the results are better than the results of the best of the two types of sensors alone.

In addition, Table 2 includes also the correct identification rate of the single sensors, which are interesting to assess the improvement to the identification introduced by the cooperation among the robots. Interestingly, the best sensors seem to be the reflectance sensors, closely followed by the cameras. As we will explain in the conclusion, we want to check this behavior with real data gathered with real sensors.

7. CONCLUSIONS

In this paper we have introduced and formalized the problem of target identification in robotic swarms. The solution that we propose relies on the naive Bayes classifier paradigm in order to distribute the computation among the team and standardize the communication messages exchanged by the

robots. We have performed an extensive simulation study in 2D in order to assess the quality of the recognition and to delineate some properties of the algorithm.

Future works will follow three main directions. From a theoretical point of view, we plan to study the properties of the algorithm computing the mean value and covariance of the final probability $p(O|Z)$. In order to ensure real-world applicability we will address the case of multiple objects in the environment and apply clusterization of the team members based on proximity estimated through communication.

The developed system will be employed in 3D simulations and experiments, which will include not only the identification phase but also the important phase of the motion of the swarm. In order to validate and improve the simulated measurements, we plan to use data produced using actual sensors and real objects. In particular, although all the used values are taken from scientific literature, we aim at validating our models for the reflectance sensors. Additional types of sensors will also be evaluated, e.g.: temperature and stiffness sensors among others.

8. REFERENCES

[1] S. Abbasi and F. Mokhtarian. Automatic view selection in multi-view object recognition. In *Pattern Recognition, 2000. Proceedings. 15th International Conference on*, volume 1, pages 13–16 vol.1, 2000.

[2] C. Arth and H. Bischof. Real-time object recognition using local features on a dsp-based embedded system. *Journal of Real-Time Image Processing*, 3:4:233–253, 2008.

[3] A. M. Baldridge, S. Hook, C. Grove, and G. Rivera. The aster spectral library version 2.0. *Remote Sensing of Environment*, 113:711–715, 2009.

[4] B. Browatzki, V. Tikhanoff, G. Metta, H. Bülthoff, and C. Wallraven. Active object recognition on a humanoid robot. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 2021–2028, May 2012.

[5] K. Derr and M. Manic. Multi-robot, multi-target particle swarm optimization search in noisy wireless environments. In *Human System Interactions, 2009. HSI '09. 2nd Conference on*, pages 81–86, May 2009.

[6] R. Diaz, S. Hallman, and C. C. Fowlkes. Detecting dynamic objects with multi-view background subtraction. In *The IEEE International Conference on Computer Vision (ICCV)*, December 2013.

[7] A. Franchi, P. Stegagno, M. Di Rocco, and G. Oriolo. Distributed target localization and encirclement with a multi-robot system. In *Intelligent Autonomous Vehicles (IAV 2010), 7th IFAC Symposium on*, pages 67–72, September 2011.

[8] J. Guthrie and K. Walsh. Non-invasive assessment of pineapple and mango fruit quality using near infra-red spectroscopy. *Australian Journal of Experimental Agriculture*, 37 (2):253–263, 1997.

[9] G. Hetzel, B. Leibe, P. Levi, and B. Schiele. 3d object recognition from range images using local feature histograms. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 2, pages II–394–II–399 vol.2, 2001.

[10] D. Jia, Z. Hou, and Z. Liu. Reflection spectral imaging technology and its application in detection of foreign materials. In *Intelligent Control and Automation, 2008. WCICA 2008. 7th World Congress on*, pages 2673–2677, June 2008.

[11] H. Liu, X. Song, J. Bimbo, L. Seneviratne, and K. Althoefer. Surface material recognition through haptic exploration using an intelligent contact sensing finger. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pages 52–57, Oct 2012.

[12] D. Lowe. Object recognition from local scale-invariant features. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, volume 2, pages 1150–1157, 1999.

[13] D. McGibney, T. Umeda, K. Sekiyama, H. Mukai, and T. Fukuda. Cooperative distributed object classification for multiple robots with audio features. In *Micro-NanoMechatronics and Human Science (MHS), 2011 International Symposium on*, pages 134–139, Nov 2011.

[14] N. Naikal, A. Yang, and S. Sastry. Towards an efficient distributed object recognition system in wireless smart camera networks. In *Information Fusion (FUSION), 2010 13th Conference on*, pages 1–8, July 2010.

[15] B. Schiele and J. L. Crowley. Recognition without correspondence using multidimensional receptive field histograms. *International Journal of Computer Vision*, 36:31–50, 2000.

[16] M. J. Swain and D. H. Ballard. Color indexing. *International Journal of Computer Vision*, 7:1:11–32, 1991.

[17] A. Thomas, V. Ferrar, B. Leibe, T. Tuytelaars, B. Schiel, and L. Van Gool. Towards multi-view object class detection. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 1589–1596, 2006.

[18] J. Wang and N. Zhang. Analysis on target recognition for soccer robots. In *Control and Decision Conference, 2008. CCDC 2008. Chinese*, pages 3273–3276, July 2008.

[19] Y. Wang, M. Brookes, and P. Dragotti. Object recognition using multi-view imaging. In *Signal Processing, 2008. ICSP 2008. 9th International Conference on*, pages 810–813, Oct 2008.

[20] B. Zhang, K. Chen, and Y. Liang. Analysis on reliability of acoustic features for material recognition of damped impacted plates. In *Information Science and Technology (ICIST), 2013 International Conference on*, pages 1641–1644, March 2013.