IEEE Robotics and Automation Letters (RAL) paper presented at the
2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 25-29, 2020, Las Vegas, NV, USA (Virtual)

Distributed Consistent Multi-Robot Semantic Localization and Mapping

Vladimir Tchuiev[1] and Vadim Indelman[2]

Abstract— **We present an approach for multi-robot consistent distributed localization and semantic mapping in an unknown environment, considering scenarios with classification ambiguity, where objects' visual appearance generally varies with viewpoint. Our approach addresses such a setting by maintaining a distributed posterior hybrid belief over continuous localization and discrete classification variables. In particular, we utilize a viewpoint-dependent classifier model to leverage the coupling between semantics and geometry. Moreover, our approach yields a consistent estimation of both continuous and discrete variables, with the latter being addressed for the first time, to the best of our knowledge. We evaluate the performance of our approach in a multi-robot semantic SLAM simulation and in a real-world experiment, demonstrating an increase in both classification and localization accuracy compared to maintaining a hybrid belief using local information only.**

I. Introduction

Deployment of multi-robot systems allow for fast information gathering, and can be used in a wide variety of applications, for example: search and rescue, autonomous driving, and agriculture. A significant part of ongoing research is multi-robot Simultaneous Localization and Mapping (SLAM), where a group of robots localize themselves and cooperatively map the environment. Multi-robot SLAM is utilized in a variety of navigation tasks such as cooperative search and rescue, underwater navigation, or warehouse management. SLAM itself is a widely researched problem (see e.g. [1]) in the robotics community. In particular, semantic SLAM reasons about objects within the environment with richer information, such as object's class, compared to geometric SLAM. Yet, often when observed from certain viewpoints, inferring the correct class of an object can be challenging, i.e. an object may visually appear similar to representative objects from different classes. This induces a viewpoint dependency for classifier outputs and requires information from different viewpoints for maintaining a belief over classification variables.

In this paper we present the first distributed multi-robot approach for semantic localization and mapping in the above setting. Our approach maintains a hybrid belief over continuous variables (object and camera poses) and discrete variables (object classes), while considering the coupling between classification and localization, and enforcing consistent, double-counting-free estimation.

In contrast, existing approaches for multi-robot semantic SLAM utilize most-likely class measurements to solve data association. Moreover, these approaches do not maintain a belief over classification variables, nor model the coupling between semantic and geometric information.

As each robot uses information from other robots, it must not use measurements more than once, otherwise it will lead to erroneous and overconfident estimates, i.e. it will double count information. To address this key problem, multiple approaches were proposed, all considering continuous variables: from complex book-keeping (e.g. [2]) to information removal techniques (e.g. [3]). In this work we address consistent inference of a hybrid belief that consists of continuous and discrete variables. To the best of our knowledge, the latter has not been addressed thus far.

To summarize, our main contributions are as follows. (i) we contribute a multi-robot approach that maintains a hybrid belief over robot and object poses, and object classes in a distributed setting, while addressing the coupling between semantic and geometric information via viewpoint-dependent classifier model; (ii) we address estimation consistency aspects considering both continuous and discrete random variables; (iii) we demonstrate the strength of this approach in simulation and real-world experiment, comparing to single robot and distributed multi-robot with double counting. This paper is accompanied with supplementary material [4] which provides further details and results.

II. Related Work

Various works have utilized sequential classification with a classifier model for a single robot. Omidshafiei et al. [5] presented a sequential classification approach that used a Dirichlet distributed classifier model. The classifier model was not modeled as viewpoint-dependent. Kopitkov and Indelman [6] presented an approach to train a viewpoint dependent classifier model. Feldman and Indelman [7] proposed a sequential object classification that utilizes a viewpoint dependent classifier with known relative poses a-priori. Tchuiev et al. [8] maintained a hybrid belief with a viewpoint dependent classifier to disambiguate between data association realizations. These works, [8], address only sequential classification and do not consider the coupled problem with SLAM. To our knowledge, our work is the first to address the coupled problem in a distributed setting.

There are different approaches for distributed multi-robot SLAM; Walls et al. [9] proposed a distributed geometric

Manuscript received: February 25th, 2020; Revised: May 12th, 2020; Accepted: June 4th, 2020.

This paper was recommended for publication by Editor Tamim Asfour upon evaluation of the Associate Editor and Reviewers' comments.

This work was partially supported by the Israel Ministry of Science & Technology (MOST).

The authors are with the Department of Aerospace Engineering, Technion - Israel Institute of Technology, Haifa 32000, Israel. {vovatch[1], vadim.indelman[2]}@technion.ac.il.

Digital Object Identifier (DOI): see top of this page.

Copyright ©2020 IEEE

SLAM approach that communicates factors between robots. Other approaches for geometric SLAM include Extended Kalman Filter (such as [10]) or Particle Filter based methods (such as [11]). Choudhary et al. [12] presented an approach for distributed semantic SLAM which communicates relative poses between robots and uses object class information for data association. The geometric approaches do not reason about object classes, while the semantic approaches consider only most likely classification, i.e. do not maintain a belief over class variables. Our semantic approach maintains a belief over object classes and considers the coupling between the continuous and discrete variables.

Consistent estimation is a key issue in a distributed setup, with multiple approaches proposed to address it. Bahr et al. [2] proposed a distributed algorithm for under-water vehicles, with an approach for using all measurements without information loss. Indelman et al. [13] proposed a graph based method that calculated cross-covariance terms that represent the correlation between measurements from different robots, utilizing it for consistent estimation. Cunningham et al. [14] presented the DDF-SAM distributed SLAM algorithm that avoided double counting by creating two maps for each robot: local and global. The global map is updated with condensed local maps. A later work by Cunningham et al. [3] introduced DDF-SAM2, where each robot maintains only the global map. To avoid double counting, the old information during communication is filtered out via down-dating by each robot. These approaches consider continuous random variables. In contrast, we reason about discrete variables as well.

III. Notations and Problem Formulation

Consider a group of robots operating in an unknown environment represented by object landmarks. All of the robots aim to localize themselves, and map the environment geometrically and semantically within a distributed multi-robot framework. In this work we consider a closed-set setting, where each of the objects is of one of M possible classes. The number of objects in the environment prior to the scenario is unknown.

We denote states inferred by robot r with a superscript $\square^r$. Set R is the set of all robots communicating with robot r (including itself), either directly, or relayed through other robots. Note that R can increase its size with time. Let x_k denote robot pose at time k, x_n^o and c_n denote the n'th object pose and class respectively. Let $\mathcal{X}^o \doteq \{x_n^o\}_n$ and $C \doteq \{c_n\}_n$ denote poses and classes of objects, and $\mathcal{X}_k \doteq \{x_{0:k}, \mathcal{X}_k^o\}$ denotes all poses up to time k. Subscript new, k representing the objects newly observed at k.

Let $\mathcal{Z}_k^r$ be the set of measurements robot r receives at time k by its own sensors. $\mathcal{Z}_k^r$ is composed of geometric and semantic measurements $\mathcal{Z}_k^{geo,r}$, and $\mathcal{Z}_k^{sem,r}$ respectively. We assume independence between geometric and semantic measurements, as well as between different time steps.

We assume Gaussian and known identical motion $\mathcal{M}_k \doteq \mathbb{P}(x_k|x_{k-1}, a_{k-1})$ and geometric $\mathbb{P}(z_k^{geo,r}|x_k^r, x^{o,r})$ models for all robots. At each time step, there is a subset of object poses involved in the geometric and classifier model that is determined by data association (DA). Unlike our previous work [8], herein, DA is assumed to be externally determined.

Parameters	
x	Robot pose
x_n^o, c_n	n'th object pose and class
$\mathcal{X}_k^o$	Poses of objects observed up to time k
$\mathcal{X}_{new,k}^o$	Poses of objects newly observed at time k
$\mathcal{X}_k$	Robot and object poses up to time k
C_k	Object seen up to time k class realization
$C_{new,k}$	Classes of objects newly observed at time k
$\mathcal{Z}_k$	Measurements at time k including geometric and semantic
$\mathcal{M}_k$	Motion model from x_{k-1} to x_k
$\mathcal{L}_k$	Measurement likelihood of $\mathcal{Z}_k$
$\mathcal{H}_k$	History of measurements and action up to time k
b_k	Conditional continuous belief at time k
w_k	Discrete weight at time k
ξ_k	Continuous object marginal belief at time k
ϕ_k	Discrete marginal belief at time k
$N_k(\cdot)$	Number of objects observed by a robot or a group up to time k
Superscripts	
r	States of robot r
R	States of robots communicating with r, directly and indirectly, including itself

TABLE I: Main notations used in the paper.

Additionally, we use a viewpoint-dependent classifier model that "predicts" classification scores (a vector of class probabilities). This model couples classifier scores with viewpoint dependency between object and camera; this coupling can be used to improve pose inference performance [8]. The viewpoint dependency is modeled as a Gaussian with parameters that depend on the relative viewpoint from the camera to the object $x^{o,r} \ominus x_k^r$ and object's class c:

$$\mathbb{P}(z_k^{sem,r}|x_k^r, x^o, c) = \mathcal{N}(h_c(x_k^r, x^{o,r}), \Sigma_c(x_k^r, x^{o,r})), \quad (1)$$

where $h_c(\cdot)$ and $\Sigma_c(\cdot)$ can be learned offline via a Gaussian Process (GP) [7] or a deep neural network [6]. Note that for M candidate classes, M viewpoint-dependent models have to be learned.

Let $\mathcal{L}_k^r \doteq \mathbb{P}(\mathcal{Z}_k^r|\mathcal{X}_k^r, C_k^r)$ be the local measurement likelihood of r that consists of geometric and classifier models:

$$\mathcal{L}_k^r \doteq \prod_{x^{o,r}, c^r} \mathbb{P}(\mathcal{Z}_k^{geo,r}|x_k^r, x^{o,r})\mathbb{P}(\mathcal{Z}_k^{sem,r}|x_k^r, x^{o,r}, c^r), \quad (2)$$

where $x^{o,r} \in \mathcal{X}_{\beta_k}^{o,r}$ and $c^r \in C_{\beta_k}^r$; the term β_k represents the local DA of robot r at time k, i.e. the correspondences between observations and object IDs. Denote $\mathcal{X}_{\beta_k}^{o,r}$ the set of all poses of objects that observed by r at time k, and similarly denote $C_{\beta_k}^r$ for object classes. For the reader's convenience, Table I presents the important notations used in the paper, some will be defined in the next section.

Problem formulation: For each robot r we aim to maintain the following hybrid belief:

$$\mathbb{P}(\mathcal{X}_k^R, C^R|\mathcal{H}_k^R), \quad (3)$$

where $\mathcal{H}_k^R \doteq \{\mathcal{Z}_{1:k}^{r'}, a_{0:k-1}^{r'}\}_{r' \in R}$ is the history of measurements of robot r itself and transmitted information to r, as well as actions from all robots in R. The belief in Eq. (3) is a hybrid belief over both continuous (camera and object poses), and discrete (object classes) random variables. We aim to update this hybrid belief per each robot in a recursive manner, using both local measurements and information sent from other robot in the neighborhood, as well as sending information by itself. We aim to keep estimation consistency by avoiding double counting, i.e. using every measurement only once.

IV. Approach

We present a framework for distributed classification, localization, and mapping. As with many multi-robot distributed frameworks, over-confident estimations, due to double counting, is a key issue; We propose a framework that simplifies the book-keeping that allows relaying of information (e.g. robot 1 sends information to robot 2, then 2 sends to 3 information that also includes the received from robot 1). This framework requires the maintenance of a local belief $\mathbb{P}(\mathcal{X}_k^r, C^r|\mathcal{H}_k^r)$ per each robot that can be sent and relayed to other robots. From multiple local beliefs a distributed belief can be constructed. The local beliefs are stored by each robot, and updated accordingly when new information arrives, and the receiving robot filters out the old information, thus avoiding double counting.

In the next sections we derive a recursive formulation for maintenance of the local belief, the distributed hybrid belief, and the information stack each robot holds and transmits.

A. Local Hybrid Belief Maintenance

Our formulation for maintaining local hybrid beliefs builds upon our previous work [8], with the main differences being that here we assume the DA is solved, and the number of objects is unknown a-priori. In this section we present an overview of this approach.

We maintain the hybrid belief of robot r only from local information. This belief can be split into continuous and discrete parts as in:

$$\mathbb{P}(\mathcal{X}_k^r, C_k^r|\mathcal{H}_k^r) = \underbrace{\mathbb{P}(\mathcal{X}_k^r|C_k^r, \mathcal{H}_k^r)}_{b_k^r}\underbrace{\mathbb{P}(C_k^r|\mathcal{H}_k^r)}_{w_k^r}. \tag{4}$$

To maintain this hybrid belief, we must maintain a set of continuous beliefs conditioned on the class realization of all objects observed in the scene by robot r thus far.

The continuous part can be updated as follows:

$$b_k^r \propto b_{k-1}^r \cdot \mathcal{L}_k^r \cdot \mathcal{M}_k^r \cdot \mathbb{P}(\mathcal{X}^{o,r}_{\text{new},k}), \tag{5}$$

where $\mathbb{P}(\mathcal{X}^{o,r}_{\text{new},k}) = \frac{\mathbb{P}(\mathcal{X}^{o,r}_k)}{\mathbb{P}(\mathcal{X}^{o,r}_{k-1})}$ is the prior over object poses newly observed at time k. As opposed to [8], this formulation also supports an increasing number of objects known at each time step, with both $\mathcal{X}_k^{o,r}$ and C_k^r increasing in dimension. Note that in general b_k^r is different for each class realization, as models (1) are different for each class.

The discrete part is the weight associated to its corresponding continuous belief. As our measurement models depend on continuous variables, we use Bayes rule on $\mathbb{P}(C_k^r|\mathcal{H}_k^r)$ and marginalize the measurement likelihood as follows:

$$w_k^r \propto w_{k-1}^r \mathbb{P}(C^r_{\text{new},k}) \int_{\mathcal{X}_k^r} \mathcal{L}_k^r \cdot b_{k-1}^r \cdot \mathcal{M}_k^r d\mathcal{X}_k^r, \tag{6}$$

where $\mathbb{P}(C^r_{\text{new},k}) = \frac{\mathbb{P}(C_k^r)}{\mathbb{P}(C_{k-1}^r)}$ is the prior over classes of new objects locally observed by r at time k. We compute the integral in Eq. (6) by sampling the continuous variables that participate in $\mathbb{P}(\mathcal{Z}_k^r|\mathcal{X}_k^r, C_k^r)$, i.e. the last robot pose x_k^r and the poses of observed objects $\mathcal{X}^{o,r}_{\beta_k}$ at time k. These variables are sampled from the propagated belief $b_{k-1}^r \cdot \mathcal{M}_k^r$. Variables that do not participate in $\mathcal{L}_k^r$ can be marginalized analytically.

B. Distributed Hybrid Belief Maintenance

In this section we extend the formulation presented in Sec. IV-A to include updates from other robots, considering a distributed multi-robot setting. As will be seen, our formulation uses each measurement only once, thus keeping estimation consistency and avoiding double counting. Similarly to (4), we factorize the distributed hybrid belief (3)

$$\mathbb{P}(\mathcal{X}_k^R, C_k^R|\mathcal{H}_k^R) = \underbrace{\mathbb{P}(\mathcal{X}_k^R|C_k^R, \mathcal{H}_k^R)}_{b_k^R}\underbrace{\mathbb{P}(C_k^R|\mathcal{H}_k^R)}_{w_k^R}. \tag{7}$$

As in the single robot case, maintaining this belief requires managing multiple hypotheses of class realizations. Compared to the single robot case, the number of objects observed will be equal or greater for distributed belief, therefore the number of possible realizations increases as well. Importantly, information transmitted by other robots impacts both b_k^R and w_k^R. Furthermore, the classifier viewpoint-dependent model induces coupling between localization uncertainty and classification of different robots.

We present a recursive formulation for maintaining each of the parts in (7). The distributed measurement history $\mathcal{H}_k^R$ can be split to a prior part, and a new part, defined as $\Delta\mathcal{H}_k^R$, that consists of measurements and actions from time k, s.t: $\mathcal{H}_k^R = \mathcal{H}_{k-1}^R \cup \Delta\mathcal{H}_k^R$. Similarly, let $\mathcal{H}_k^r \doteq \mathcal{H}_{k-1}^r \cup \{\mathcal{Z}_k^r, a_{k-1}^r\}$ for the single robot case. Note information in $\Delta\mathcal{H}_k^R$ transmitted by other robots can potentially be from earlier time instances (as each robot during communication transmits to robot r its own stack of local beliefs of other robots, see Section IV-C). Crucially, each measurement must be used once to avoid double counting. We also denote history *without* local measurements and action at time k as

$$\mathcal{H}_k^{R-} \doteq \mathcal{H}_k^R \setminus \{\mathcal{Z}_k^r, a_{k-1}^r\}, \; \Delta\mathcal{H}_k^{R-} \doteq \Delta\mathcal{H}_k^R \setminus \{\mathcal{Z}_k^r, a_{k-1}^r\}. \tag{8}$$

Using the above notations, one can observe $\mathcal{H}_k^{R-} = \mathcal{H}_{k-1}^R \cup \Delta\mathcal{H}_k^{R-}$. Next, we detail our approach for maintaining both the conditional continuous part b_k^R and the discrete part w_k^R recursively for a realization of object classes C_k^R.

1) Maintaining b_k^R: Using Bayes rule, we rewrite b_k^R as:

$$b_k^R = \eta \cdot \mathcal{L}_k^r \cdot b_k^{R-} \tag{9}$$

where $\eta \doteq \mathbb{P}(\mathcal{Z}_k^r|C_k^r, \mathcal{H}_k^R \setminus \mathcal{Z}_k^r)^{-1}$ is a normalization constant the does not participate in inference of the continuous belief. The local measurement likelihood, $\mathcal{L}_k^r$, is defined in Eq. (2).

The term $b_k^{R-} \doteq \mathbb{P}(\mathcal{X}_k^R|C_k^R, \mathcal{H}_k^R \setminus \mathcal{Z}_k^r)$ is the distributed propagated belief that is conditioned on information transmitted by other robots at time k, and on the latest action of robot r but not on its local measurement. During update, b_k^{R-} is saved to be used in maintenance of w_k^R, as seen in the next subsection. Using chain rule, we can extract the motion model of the latest action as well:

$$b_k^{R-} = \mathcal{M}_k^r \cdot \mathbb{P}(\mathcal{X}_k^R \backslash x_k^r | C_k^R, \mathcal{H}_k^{R-}). \tag{10}$$

We can express $\mathbb{P}(\mathcal{X}_k^R \backslash x_k^r | C_k^R, \mathcal{H}_k^{R-})$ in terms of the distributed continuous prior $b_{k-1}^R \doteq \mathbb{P}(\mathcal{X}_{k-1}^R | C_{k-1}^R, \mathcal{H}_{k-1}^R)$, and the new information received from other robots (see [4, Sec. 2]):

$$\mathbb{P}(\mathcal{X}_k^R \backslash x_k^r | C_k^R, \mathcal{H}_k^{R-}) = b_{k-1}^R \cdot \frac{\mathbb{P}(\mathcal{X}_k^{o,R} | C_k^{o,R}, \Delta\mathcal{H}_k^{R-})}{\mathbb{P}(\mathcal{X}_{k-1}^{o,R})} \tag{11}$$

Finally, we substitute Eq. (11) to Eq. (10) and in turn to Eq. (9), and get the following recursive formulation:

$$b_k^R \propto b_{k-1}^R \cdot \mathcal{L}_k^r \cdot \mathcal{M}_k^r \cdot \mathbb{P}(\mathcal{X}_{\text{new},k}^{o,R}) \frac{\mathbb{P}(\mathcal{X}_k^{o,R} | C_k^{o,R}, \Delta\mathcal{H}_k^{R-})}{\mathbb{P}(\mathcal{X}_k^{o,R})}, \tag{12}$$

where the measurement likelihood $\mathcal{L}_k^r$ accounts for the new local measurement, $\mathcal{M}_k^r$ accounts for the latest action of robot r, and $\mathbb{P}(\mathcal{X}_k^{o,R} | C_k^{o,R}, \Delta\mathcal{H}_k^{R-})$ (shown in blue) accounts for new information sent to r by other robots in R at time k. This pdf is only over object poses ($\mathcal{X}_k^{o,R}$), while the other robots' poses are marginalized out. Thus, robots communicate the environment states, which are implicitly affected by the robots' pose estimation. Computation of the blue part is further discussed in Sec. IV-C. Compared to the local belief update (5), the blue part is the main difference. The expression $\mathbb{P}(\mathcal{X}_{\text{new},k}^{o,R})$ represents pose prior of objects newly known by r at time k.

The distributed belief has at worst $M^{N_k(R)}$ continuous beliefs with corresponding weights, where the number of objects $N_k(R)$ known by r can increase with time. Naturally, a multi-robot system will observe more objects than a single robot, therefore the computational burden for distributed belief will be larger than for the local belief. Therefore, the significance of pruning beliefs with small weight grows. We set a threshold for the ratio between a weight and the largest weight in the distributed hybrid belief.

2) Maintaining w_k^R: To maintain w_k^R, we use a similar derivation to the weight update via local information only, presented in Sec. IV-A. We use Bayes rule to extract the last local measurement likelihood:

$$w_k^R = \eta \cdot w_k^{R-} \cdot \mathbb{P}(\mathcal{Z}_k^r | C_k^R, \mathcal{H}_k^R \backslash \mathcal{Z}_k^r), \tag{13}$$

where $w_k^{R-} \doteq \mathbb{P}(C_k^R | \mathcal{H}_k^R \backslash \mathcal{Z}_k^r)$ is the posterior distributed weight without accounting for the latest local measurements, and $\eta \doteq \mathbb{P}(\mathcal{Z}_k^r | \mathcal{H}_k^R \backslash \mathcal{Z}_k^r)^{-1}$ is a normalization constant that is identical in all realizations of C_k^R, thus does not participate in weight inference. As we use a viewpoint dependent classifier model that utilizes the coupling between relative viewpoint and object class, we need to marginalize $\mathbb{P}(\mathcal{Z}_k^r | C_k^R, \mathcal{H}_k^R \backslash \mathcal{Z}_k^r)$ over the involved poses in this likelihood: the last robot pose x_k^r, and poses of objects observed at time k. We denote the latter by $\mathcal{X}_{\beta_k}^{o,r}$, and to shorten notations denote $\mathcal{X}_{\text{inv},k}^r \doteq \{x_k^r, \mathcal{X}_{\beta_k}^{r,k}\}$, and by $\neg\mathcal{X}_{\text{inv},k}^r$. Thus, $\mathbb{P}(\mathcal{Z}_k^r | C_k^R, \mathcal{H}_k^R \backslash \mathcal{Z}_k^r)$ is marginalized as

$$\mathbb{P}(\mathcal{Z}_k^r | C_k^R, \mathcal{H}_k^R \backslash \mathcal{Z}_k^r) = \int_{\mathcal{X}_{\text{inv},k}^r} \mathcal{L}_k^r \cdot \mathbb{P}(\mathcal{X}_{\text{inv},k}^r | C_k^r, \mathcal{H}_k^R \backslash \mathcal{Z}_k^r) d\mathcal{X}_{\text{inv},k}^r, \tag{14}$$

where $\mathbb{P}(\mathcal{X}_{\text{inv},k}^r | C_k^r, \mathcal{H}_k^R \backslash \mathcal{Z}_k^r)$ is computed by marginalizing b_k^{R-} over the uninvolved variables $\neg\mathcal{X}_{\text{inv},k}^r$, with $\mathcal{X}_k^R = \mathcal{X}_{\text{inv},k}^r \cup \neg\mathcal{X}_{\text{inv},k}^r$, as

$$\mathbb{P}(\mathcal{X}_{\text{inv},k}^r | C_k^r, \mathcal{H}_k^R \backslash \mathcal{Z}_k^r) = \int_{\neg\mathcal{X}_{\text{inv},k}^r} b_k^{R-} d\neg\mathcal{X}_{\text{inv},k}^r. \tag{15}$$

The propagated distributed belief b_k^{R-} is given to us from the continuous belief with Eq. (10), and includes the external information, shown in blue.

In practice, we sample the involved variables $\mathcal{X}_{\text{inv},k}^r$ in the current measurement likelihood and compute its value. As b_k^R and $\mathcal{L}_k^r$ are Gaussian, η does not play a role in the sampling process. Despite the classifier outputs being modeled as Gaussian, we integrate over poses; In the general case, expectation and covariance of the classifier model are a function of the relative viewpoint, thus we need to sample $\mathcal{X}_{\text{inv},k}^r$ as presented in Sec. IV-A at Eq. (6).

The other term we will address from Eq. (13) is w_k^{R-}. We express w_k^{R-} in terms of w_{k-1}^R:

$$w_k^{R-} \propto w_{k-1}^R \cdot \mathbb{P}(C_{k-1}^R)^{-1} \cdot \mathbb{P}(C_k^R | \Delta\mathcal{H}_k^R \backslash \mathcal{Z}_k^r). \tag{16}$$

Finally, we substitute Eq. (14) and (16) to Eq. (13) to reach our final recursive form for the discrete belief update:

$$w_k^R \propto w_{k-1}^R \cdot \mathbb{P}(C_{\text{new},k}^R) \frac{\mathbb{P}(C_k^R | \Delta\mathcal{H}_k^R \backslash \mathcal{Z}_k^r)}{\mathbb{P}(C_k^R)} \int_{\mathcal{X}_{\text{inv},k}^r} \mathcal{L}_k^r \cdot \\ \cdot \mathbb{P}(\mathcal{X}_{\text{inv},k}^r | C_k^r, \mathcal{H}_k^R \backslash \mathcal{Z}_k^r) d\mathcal{X}_{\text{inv},k}^r, \tag{17}$$

with $\mathbb{P}(\mathcal{X}_{\text{inv},k}^r | C_k^r, \mathcal{H}_k^R \backslash \mathcal{Z}_k^r)$ computed via Eq. (15). This is a recursive formulation that includes the discrete prior w_{k-1}^R, external updates for the class probability from other robots (shown in red), and the external updates for the continuous belief contained within the integral.

Remark: One might be tempted to infer the class of each object separately, but it is not accurate due to the coupling between relative viewpoint and object class, as each object class is possibly implicitly dependent on all poses: robot and objects (see [4, Sec. 3]).

C. Communication Between Robots

In Sec. IV-B we presented a framework to maintain a hybrid belief of r given information obtained from other robots in R. That information was represented by the continuous blue expression in Eq. (12) and implicitly in Eq. (17), and the discrete red expression in Eq. (17). In this section, we present our approach for computing these parts, thus describing the management of this information and what each robot sends when communicating. We aim to achieve two goals:

1) Simple double counting prevention when maintaining the distributed belief without complex bookkeeping.
2) Distributed belief inference also via data not directly transmitted (e.g. robot r_1 sends data to r_2, r_2 to r_3, and r_3 is using data from r_1).

As will be shown next, the blue and red terms in Eqs. (12) and (17) can be expressed via local information transmitted by different robots in R to robot r. To that end, each robot r maintains and broadcasts a *stack* of local hybrid beliefs of other robots it is aware of. In contrast to (4), these local beliefs are marginal beliefs over object poses and classes, i.e. robot poses are marginalized out. Each slot for robot r' in the stack of robot r contains $N_k(r')$ continuous and discrete marginal beliefs (defined below as $\xi_k^{r,r'}$ and $\phi_k^{r,r'}$), one pair per class realization, following a factorization similar to (4). Additionally, each slot includes a time-stamp that indicates on what data the local hybrid belief is conditioned upon. All in all, every stack contains $\sum_{i=1}^{|R|} N_k(r_i)$ continuous and discrete beliefs. Eq. (18) presents the stack of robot r as a set of slots, where each slot contains a hybrid belief of a particular robot $r_i \in R$ over object poses and classes, normalized by their priors.

$$\mathcal{S}_k^r \doteq \left\{ \left(\frac{\mathbb{P}(\mathcal{X}_{k_i}^{o,r_i}|C_{k_i}^{r_i}, \mathcal{H}_{k_i}^{r_i})\mathbb{P}(C_{k_i}^{r_i}|\mathcal{H}_{k_i}^{r_i})}{\mathbb{P}(\mathcal{X}_{k_i}^{o,r_i})\mathbb{P}(C_{k_i}^{r_i})}, k_i \right) \right\}_{r_i \in R}, \quad (18)$$

where k_i is the time-stamp when robot r received information about r_i. In general, time k_i is not synchronized with k. The marginal continuous and discrete beliefs that robot r has about robot $r_i \in R$ are denoted $\xi_k^{r,r_i} \doteq \mathbb{P}(\mathcal{X}_{k_i}^{o,r_i}|C_{k_i}^{r_i}, \mathcal{H}_{k_i}^{r_i})/\mathbb{P}(\mathcal{X}_{k_i}^{o,r_i})$ for the continuous part, and $\phi_k^{r,r_i} \doteq \mathbb{P}(C_{k_i}^{r_i}|\mathcal{H}_{k_i}^{r_i})/\mathbb{P}(C_{k_i}^{r_i})$ for the discrete part.

With these definitions of ξ_k^{r,r_i} and ϕ_k^{r,r_i}, it is possible to show that the blue part in Eq. (12) can be expressed as (see full derivation in supplementary material [4, Sec. 4])

$$\frac{\mathbb{P}(\mathcal{X}_k^{o,R}|C_k^R, \Delta\mathcal{H}_k^{R-})}{\mathbb{P}(\mathcal{X}_k^{o,R})} = \prod_{r_i \in R} \frac{\xi_k^{r,r_i}}{\xi_{k-1}^{r,r_i}} \quad (19)$$

Similarly, the red term in Eq. (17) can be expressed as (see full derivation in supplementary material [4, Sec. 5]):

$$\frac{\mathbb{P}(C_k^R|\Delta\mathcal{H}_k^R \backslash \mathcal{Z}_z^r)}{\mathbb{P}(C_k^R)} = \prod_{r_i \in R} \frac{\phi_k^{r,r_i}}{\phi_{k-1}^{r,r_i}}. \quad (20)$$

Eqs. (19) and (20) present the external update as a product of local beliefs, with only the updates from $k-1$ for robot r are present. This formulation avoids double counting by removing old information, ξ_{k-1}^{r,r_i} and ϕ_{k-1}^{r,r_i}, in each communication and uses measurements only once. Specifically for ξ_{k-1}^{r,r_i}, we use the approach presented in [3]. Doing so by maintaining stacks of individual information does not require complex book-keeping, only time-stamps for each slot; Thus we fulfill the first goal. Robots can also relay information transmitted to them, thus the distributed belief can be updated by information from robots that did not transmit to the inferring robot, thus fulfilling the second goal.

Robot r_i sends the entire stack during information broadcast. When robot r receives information, it integrates the broadcast in as follows: recall that r_i's stack is divided to slots, with a time stamp per each slot. Robot r compares time stamps with the received information per slot, and replaces the information within the slot if the received time stamps is newer. If r receives information from more than one other robot at the same time, it will select the newest information per slot. This procedure is dependent on the relations between time-stamps, thus it is not necessary to synchronize time between the robots.

In the following section we discuss double counting aspects of discrete random variables, corresponding to Eq. (20).

D. Double Counting of Discrete Random Variables

Double counting leads to over-confident estimations, and if an erroneous measurement is counted multiple times, it may lead to a large error in the state's estimation in turn. While the implications of double counting on continuous random variables (e.g. camera poses and objects) have been investigated, it is not so for discrete random variables. Both cases have a common thread: measurements counted multiple times will 'push' the posterior estimation to a certain direction while leading to lower uncertainty than when double counting is appropriately avoided (i.e. each measurement is used at most once). In the continuous Gaussian case, it manifests in a covariance matrix with smaller eigenvalues. Comparatively, in the discrete case the highest probability category will have its probability increase while the probability of not being in this category decreases.

To illustrate the above, consider an example with a categorical random variable c; we receive two sets of data $Z_a = \{z_1, z_2\}$, and $Z_b = \{z_2, z_3\}$, with a common measurement z_2. Considering a measurement likelihood $\mathbb{P}(z|c)$, the posterior over c is (see e.g. Bailey et al. [15]):

$$\mathbb{P}(c|Z_a, Z_b) \propto \mathbb{P}(c)\mathbb{P}(Z_a, Z_b|c) = \mathbb{P}(c)\frac{\mathbb{P}(z_1|c)\mathbb{P}(z_2|c)^2\mathbb{P}(z_3|c)}{\mathbb{P}(z_2|c)}. \quad (21)$$

If the common data (measurement z_2) is not removed via the denominator in Eq. (21), it will be double counted. Compared to Eq. (20), the above nominator and denominator correspond, respectively, to the terms ϕ_k^{r,r_i} and ϕ_{k-1}^{r,r_i}.

Denote $\mathbb{P}(z_2|c = i) \doteq a_i$, and to shorten the notations $\mathbb{P}(c = i)\mathbb{P}(z_1|c = i)\mathbb{P}(z_3|c = i) \doteq \mathcal{L}_i$. The normalized posterior can be written as:

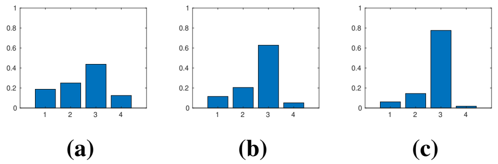
$$\mathbb{P}(c = i|Z_a, Z_b) = \frac{a_i\mathcal{L}_i}{\sum_{j=1}^m a_j\mathcal{L}_j} = \frac{a_i^2\mathcal{L}_i}{\sum_{j=1}^m a_j\mathcal{L}_j \cdot a_i} \quad (22)$$

where m is the number of candidate categories. Double counting, i.e. without the denominator in Eq. (21), gives after normalization $\frac{a_i^2\mathcal{L}_i}{\sum_{j=1}^m a_j^2\mathcal{L}_j}$.

The largest a_i is denoted a_{max}, with i_{max} being the category corresponding to a_{max}, and subsequently the product of all other terms for i_{max} is denoted $\mathcal{L}_{max}$. Double counting of $\mathbb{P}(z_2|c_i)$ will increase the probability of i_{max}:

$$\mathbb{P}(c = i_{max}|Z_a, Z_b) = \frac{a_{max}^2\mathcal{L}_{max}}{\sum_{j=1}^m a_j\mathcal{L}_j \cdot a_{max}} \leq \frac{a_{max}^2\mathcal{L}_{max}}{\sum_{j=1}^m a_j^2\mathcal{L}_j}. \quad (23)$$

Similarly, it can be shown that with higher power (i.e. counting the data more) can increase the posterior probability even further; In addition, the reverse can be shown for the lowest probability in a. This increase in influence can be disastrous if the category of the highest probability likelihood is not correct, possibly leading to pruning of the correct class hypothesis when maintaining the hybrid belief (3).



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

Fig. 1: Conceptual demonstration of the effects of double counting on discrete random variables. Consider 4 possible categories with an uninformative prior over them. **(a)** is the measurement likelihood for the categories. Considering the uninformative prior, it is the posterior distribution as well. **(b)** and **(c)** counts the same likelihood twice and thrice respectively.

A visualization can be seen in Fig. 1, where there are 4 categories with uninformed prior and a measurement likelihood; in Figs. 1a, 1b and 1c the likelihood is counted once, twice and thrice respectively. Evidently, the strongest category's probability (cat. 3) is increased when counted more times while all other have their probability diminish.

V. Experiments

We evaluated our approach in a multi-robot SLAM simulation and with real-world data where we consider an environment comprising several scattered objects observed by multiple mobile cameras from different viewpoints. Fig. 2a and Fig. 5a present the ground truth for simulation and experiment respectively. Our implementation uses the GTSAM library [16] with a python wrapper. The hardware used is an Intel i7-7700 processor running at 2.8GHz and 16GB RAM, with GeForce GTX 1050Ti with 4GB RAM.

A. Simulation Setting, Compared Approaches and Metrics

Consider 3 robots, denoted r_1, r_2, and r_3, moving in a 2D environment represented by $N = 15$ scattered objects. We consider a closed-set setting and assume, for simplicity, $M = 2$ classes, where each object can be one of the two. In this scenario the maximum number of possible class realizations is $M^N = 32768$.

Our approach is evaluated for both classification, and pose inference accuracy, as we maintain a hybrid belief. We consider an ambiguous scenario where the classifier model cannot distinguish between the two classes from a certain viewpoint, thus requiring additional viewpoints to correctly disambiguate between the two classes. The robots communicate between themselves, increasing performance for discrete and continuous variables, i.e. classification and SLAM. Additionally, the distributed setting extends the sensing horizon, allowing robots to reason about objects that are not directly observed, while keeping estimation consistency.

Each robot only communicates with robots within a 10 meter communication range, relaying the local information stored in its stack. In particular, initially r_2 and r_3 share information with each other, then r_1 and r_2, relaying information from r_3 through r_2. For a complete table of communication in the considered scenario, see [4, Sec. 7]. Further, we assume the robots share a common reference frame (this assumption can be relaxed as in [17]). We simulate relative pose odometry and geometric measurements, and we crafted a classifier model that simulates perceptual aliasing.

In the evaluation we compare between three approaches: local estimations, our approach, and our approach with double counting, i.e. $\xi^{r,r_i}_{k-1} = 1$ and $\phi^{r,r_i}_{k-1} = 1$ in Eq. (19) and (20) respectively. In all benchmarks we average the results for each robot. The parameters are presented in the supplementary material [4, Sec. 6].

As explained in Sec. IV-D, when double counting occurs, the posterior class probability will converge to extreme results quicker, and may result on either completely right or wrong classifications. Therefore, reasoning about a single run is insufficient, and a statistical study is required. To quantify classification accuracy, we sample 100 times different geometric and semantic measurements, and perform a statistical study over the results. For that, we use mean square detection error (MSDE) averaged over all objects, robots, and runs (also used by Teacy et al. [18] and Feldman & Indelman [7]). We define MSDE per robot and object as follows:

$$MSDE \doteq \frac{1}{m}\sum_{i=1}^{m}(\mathbb{P}_{gt}(c=i) - \mathbb{P}(c=i|\mathcal{H}^R_k))^2, \qquad (24)$$

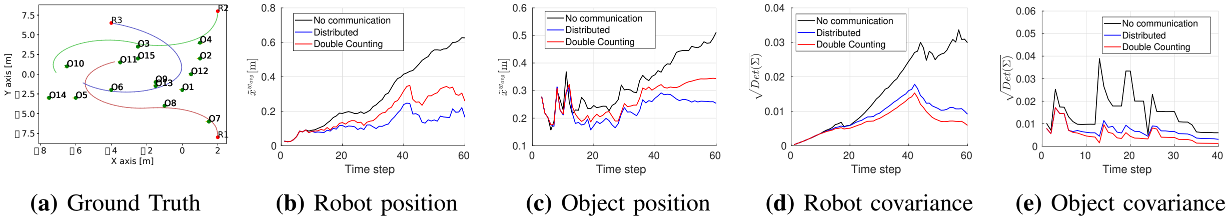
where $\mathbb{P}_{gt}(c = i)$ represents the classification ground truth and can be either 1 for the correct class or 0 for all other classes. Therefore $MSDE = 1$ for completely incorrect classification, thus allowing us to perform statistical study of the effects of double counting of discrete random variables. To quantify localization accuracy, we use estimation error $\tilde{x}^{w_{avg}}$ which is the weighted average of Euclidean distance between the estimated and ground truth poses.

B. Simulation Results

Fig. 2 presents results for continuous variables, i.e. robot and object poses. Figs. 2b and 2c show a clear advantage to our approach, where the localization error is the smallest for robots and objects respectively after the first 10 time steps. In Figs. 2d and 2e the estimation covariance is presented, where the double counted approach has the smallest values as expected. Fig. 2e shows 'spikes' in the average objects' position covariance; these correspond to new object detections where the localization uncertainty is still high.

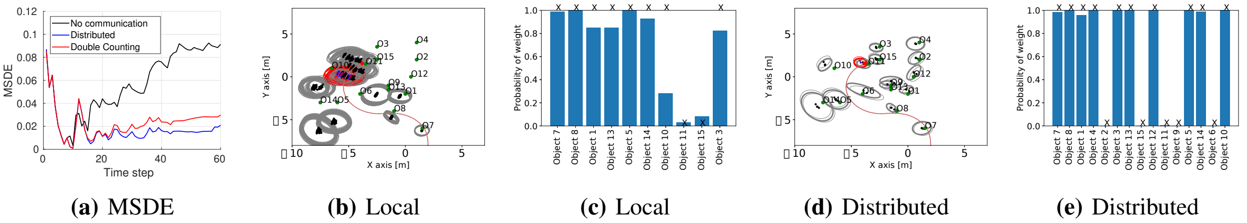
Fig. 3 visualizes classification and estimations at time $k = 60$ for local only and for distributed beliefs of robot r_2. At that time, robot r_2 communicated earlier with r_3, and for the first time communicates with r_1. When comparing Fig. 3b (local) to Fig. 3d (distributed), the number of possible class realizations is reduced. In addition, the estimate of r_2's pose, as well as the objects, is more certain and accurate. When comparing Figs. 3c and 3e, the latter presents a larger map, i.e. more objects observed, and the class estimations (classification) are closer to the ground truth.

Fig. 3a presents the average MSDE over 100 runs, where as a whole our approach shows lower MSDE values, i.e. statistically stronger classification results. In supplementary material [4, Sec. 8] we present additional classification and SLAM results.



(a) Ground Truth **(b)** Robot position **(c)** Object position **(d)** Robot covariance **(e)** Object covariance

Fig. 2: Simulation figures; **(a)** present the ground truth of the scenario. Red points represent the initial position of the robots, with different colored lines represent different robots. The green points represent the object poses. **(b)** and **(c)** represent the average $\tilde{x}^{w_{\text{avg}}}$ for robot and object position respectively as a function of time. **(d)** and **(e)** present the corresponding square-root of the position covariance for the robot and object average respectively.



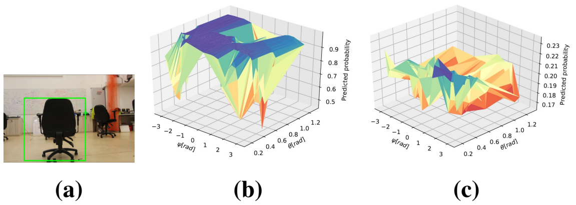
(a) MSDE **(b)** Local **(c)** Local **(d)** Distributed **(e)** Distributed

Fig. 3: **(a)** presents average MSDE for the robots over 100 runs with different measurements. The rest are figures for time $k = 60$ of r_1. **(b)** and **(d)** represent multiple SLAM hypotheses for local and distributed setting respectively; Black dots with gray ellipse represent object pose estimation, red & blue signs with red ellipse represent robot pose estimation. Green and red points represent ground truth for object and robot positions respectively. **(c)** and **(e)** represent class probabilities for $c = 1$ for objects observed thus far for local and distributed respectively. The X notations represent ground truth (1 for class $c = 1$, 0 for class $c = 2$).

C. Experiment Setting

In our scenario 3 robots are moving within an environment with multiple objects within it. We scattered 6 chairs within the environment and photographed them using a camera on a stand, keeping a constant height. In Fig. 4a we show an image from the scenario with the corresponding bounding box. The chairs were detected with YOLO3 DarkNet detector [19], which provided bounding boxes, and then each bounding box was classified using a ResNet50 convolutional neural network [20]. We considered 3 candidate classes out of 1000: 'barber chair', 'punching bag', and 'traffic light', as $c = 1, 2, 3$ respectively with $c = 1$ being the ground truth class. We trained three viewpoint-dependent classifier models using three sets of relative pose and class probability vector pairs, with the spatial parameters being the yaw and pitch angles from camera to object; The models are presented in the supplementary material [4, Sec. 9]. For the ground truth class we photographed an objects from multiple viewpoints, and then classified it using ResNet 50. For the other two classifier models, we sampled class probability vectors with larger probability for the corresponding class of the model, and used the same relative poses as the first model. Fig. 4b, 4c presents expectation of $c = 1$ for two of the classifier models as a function of the spatial parameters.

In the experiment (deployment phase), we utilized both geometric and semantic measurements, using the corresponding (learned) measurement likelihood models. Relative pose geometric measurements for odometry and between camera and objects were generated by corrupting ground truth with Gaussian noise, while the semantic measurements are provided by YOLO3 and ResNet from real images. For parameter details, see supplementary material [4, Sec. 9]. The same metrics as the simulation are used here.



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

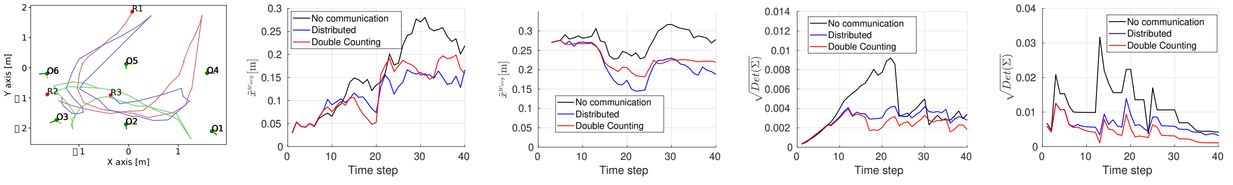
Fig. 4: **(a)** is an image used in the experiment, with corresponding the bounding box. **(b)** and **(c)** are class probability expectation for class $c = 1$ for classifier models of $c = 1$ and $c = 2$ respectively.

D. Experimental Results

Fig. 5 presents SLAM results for the same benchmarks as in Fig. 2. Figs. 5b and 5c present an average $\tilde{x}^{w_{\text{avg}}}$ over all robots for robot and object positions, respectively. In general, the advantage of our approach is evident with lower errors. In addition, Figs. 5d and 5e present a similar pattern to Figs. 2d and 2e, respectively, where the covariance of our approach is smaller than the single robot case, but larger than the over-confident double counting case.

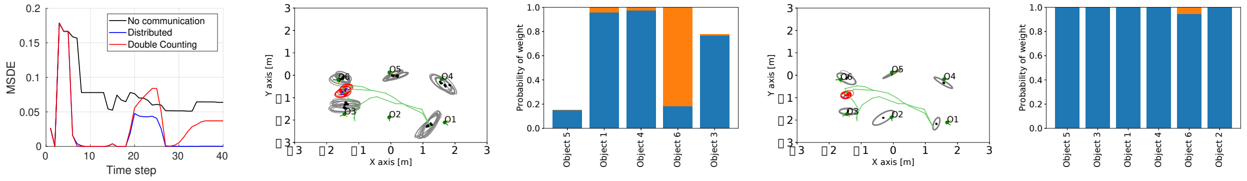
For classification results, Fig. 6a shows the average MSDE per robot as a function of time step, where eventually our approach out-performs both the single robot and the double counting cases, with higher probability for the correct class realization. In Fig. 6, SLAM and classification results for Robot 2 at time step $k = 35$ are presented, showing similar resulting trends to Fig. 3. Comparing Fig. 6b and Fig. 6d, the later shows more accurate SLAM compared to the former, with less class realizations. In addition, compared to Fig. 6e, Fig. 6c shows more accurate classification with an additional object classified.

For additional results at different time steps, refer to the supplementary material [4, Sec. 10-11] and multimedia submission.



(a) Ground Truth **(b)** Robot position **(c)** Object position **(d)** Robot covariance **(e)** Object covariance

Fig. 5: Experiment figures; **(a)** present the ground truth of the scenario. Red points represent the initial position of the robots, with different colored lines represent different robots. The green points represent the object poses. **(b)** and **(c)** represent the average $\tilde{x}^{w\text{avg}}$ for robot and object positions respectively as a function of time for the experiment. **(d)** and **(e)** present the corresponding square-root of the position covariance for the robot and object average respectively.



(a) MSDE **(b)** Local **(c)** Local **(d)** Distributed **(e)** Distributed

Fig. 6: **(a)** presents average MSDE for the robots over 100 runs with different measurements. The rest are figures for time $k = 35$ of r_2. **(b)** and **(d)** represent multiple SLAM hypotheses for local and distributed setting respectively; Black dots with gray ellipse represent object pose estimation, red & blue signs with red ellipse represent robot pose estimation. Green and red points represent ground truth for object and robot poses respectively. **(c)** and **(e)** represent class probabilities for $c = 1$ and $c = 2$ for objects observed thus far for local and distributed respectively, with blue and orange for classes 1 and 2 respectively. In this case, the ground truth class of all objects is $c = 1$.

VI. Conclusions

We presented an approach for multi-robot semantic SLAM in an unknown environment. In this approach a distributed hybrid belief is maintained per robot using local information transmitted to other robots as a 'stack', designed to keep estimation consistency without complex book-keeping, both for continuous and discrete states. We utilized a viewpoint dependent classifier model to account for the coupling of relative pose between robot and object, and object's class. In simulation and real-world experiment we showed that our approach improves classification and localization performance while avoiding double counting. Future work will incorporate data association disambiguation.

References

[1] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. D. Reid, and J. J. Leonard, "Simultaneous localization and mapping: Present, future, and the robust-perception age," *IEEE Trans. Robotics*, vol. 32, no. 6, pp. 1309 – 1332, 2016.

[2] A. Bahr, M. Walter, and J. Leonard, "Consistent cooperative localization," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, May 2009, pp. 3415–3422.

[3] A. Cunningham, V. Indelman, and F. Dellaert, "DDF-SAM 2.0: Consistent distributed smoothing and mapping," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, Karlsruhe, Germany, May 2013.

[4] V. Tchuiev and V. Indelman, "Semantic distributed multi-robot classification, localization, and mapping with a viewpoint dependent classifier model - supplementary material," Technion - Israel Institute of Technology, Tech. Rep., 2020. [Online]. Available: https://indelman.github.io/ANPL-Website/Publications/Tchuiev20ral_supplementary.pdf

[5] S. Omidshafiei, B. T. Lopez, J. P. How, and J. Vian, "Hierarchical bayesian noise inference for robust real-time probabilistic object classification," *arXiv preprint arXiv:1605.01042*, 2016.

[6] D. Kopitkov and V. Indelman, "Robot localization through information recovered from cnn classificators," in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*. IEEE, October 2018.

[7] Y. Feldman and V. Indelman, "Bayesian viewpoint-dependent robust classification under model and localization uncertainty," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2018.

[8] V. Tchuiev, Y. Feldman, and V. Indelman, "Data association aware semantic mapping and localization via a viewpoint-dependent classifier model," in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2019.

[9] J. M. Walls, A. G. Cunningham, and R. M. Eustice, "Cooperative localization by factor composition over a faulty low-bandwidth communication channel," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2015.

[10] S. Roumeliotis and G. Bekey, "Distributed multi-robot localization," *IEEE Trans. Robot. Automat.*, August 2002.

[11] A. Howard, "Multi-robot simultaneous localization and mapping using particle filters," *Intl. J. of Robotics Research*, vol. 25, no. 12, pp. 1243–1256, 2006. [Online]. Available: http://cres.usc.edu/cgi-bin/print_pub_details.pl?pubid=514

[12] S. Choudhary, L. Carlone, C. Nieto, J. Rogers, H. I. Christensen, and F. Dellaert, "Distributed mapping with privacy and communication constraints: Lightweight algorithms and object-based models," *Intl. J. of Robotics Research*, vol. 36, no. 12, pp. 1286–1311, 2017.

[13] V. Indelman, P. Gurfil, E. Rivlin, and H. Rotstein, "Graph-based distributed cooperative navigation for a general multi-robot measurement model," *Intl. J. of Robotics Research*, vol. 31, no. 9, August 2012.

[14] A. Cunningham, M. Paluri, and F. Dellaert, "DDF-SAM: Fully distributed slam using constrained factor graphs," in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2010.

[15] T. Bailey, S. Julier, and G. Agamennoni, "On conservative fusion of information with unknown non-gaussian dependence," in *Intl. Conf. on Information Fusion, FUSION*, 2012, pp. 1876 – 1883.

[16] F. Dellaert, "Factor graphs and GTSAM: A hands-on introduction," Georgia Institute of Technology, Tech. Rep. GT-RIM-CP&R-2012-002, September 2012.

[17] V. Indelman, E. Nelson, J. Dong, N. Michael, and F. Dellaert, "Incremental distributed inference from arbitrary poses and unknown data association: Using collaborating robots to establish a common reference," *IEEE Control Systems Magazine (CSM), Special Issue on Distributed Control and Estimation for Robotic Vehicle Networks*, vol. 36, no. 2, pp. 41–74, 2016.

[18] W. Teacy, S. J. Julier, R. De Nardi, A. Rogers, and N. R. Jennings, "Observation modelling for vision-based target search by unmanned aerial vehicles," in *Intl. Conf. on Autonomous Agents and Multiagent Systems (AAMAS)*, 2015, pp. 1607–1614.

[19] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.

[20] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.