Robot-Centric Perception of Human Groups

ANGELIQUE TAYLOR, University of California San Diego, USA
DARREN M. CHAN, University of California San Diego, USA
LAUREL D. RIEK, University of California San Diego, USA

The robotics community continually strives to create robots that are deployable in real-world environments. Often, robots are expected to interact with human groups. To achieve this goal, we introduce a new method, the Robot-Centric Group Estimation Model (RoboGEM), which enables robots to detect groups of people. Much of the work reported in the literature focuses on dyadic interactions, leaving a gap in our understanding of how to build robots that can effectively team with larger groups of people. Moreover, many current methods rely on exocentric vision, where cameras and sensors are placed externally in the environment, rather than onboard the robot. Consequently, these methods are impractical for robots in unstructured, human-centric environments, which are novel and unpredictable. Furthermore, the majority of work on group perception is supervised, which can inhibit performance in real-world settings. RoboGEM addresses these gaps by being able to predict social groups solely from an egocentric perspective using RGB-D data. To achieve group predictions, RoboGEM leverages joint motion and proximity estimations. We evaluated RoboGEM against a challenging, egocentric, real-world dataset where both pedestrians and the robot are in motion simultaneously, and show RoboGEM outperformed two state-of-the-art supervised methods in detection accuracy by up to 30%, with a lower miss rate. Our work will be helpful to the robotics community, and serve as a milestone to building unsupervised systems that will enable robots to work with human groups in real-world environments.

1 INTRODUCTION

Since its inception, the human-robot interaction (HRI) community has strived to design robots for real-world environments [55, 107, 114, 125, 130]. For example, robots are being used to motivate older adults to exercise to improve their health, assist clinicians with daily tasks, support workers in manufacturing, and help people navigate in airports [1, 28, 54, 54, 72, 79, 108, 115, 127, 132]. In these environments, robots are often tasked with interacting with groups of people. Thus, it is important that robots have an adequate understanding of social groups [22, 30, 44, 59, 61, 62, 66, 77, 79, 83, 84, 90, 96, 130, 133, 134].

Much prior work focuses on dyadic interaction (i.e., one human and one robot), in controlled environments that do not represent real-world conditions [51, 125]. Additionally, many methods are designed for surveillance applications, which rely on cameras placed externally in the environment. However these methods can be impractical or unfeasible, because mobile robots can often be tasked to operate in unstructured environments where they must rely solely on their onboard sensors. Moreover, exo-centric group detection and tracking methods in robotics applications can raise significant privacy concerns [16, 17, 67, 88, 102].

While nascent, there is a growing body of literature on group perception methods in HRI [75, 76, 83, 84, 86, 129, 130, 133, 134]. However, many of these approaches rely on supervised learning, which requires training models on large datasets [75, 76, 83, 84, 86]. There are a few ego-centric group detection methods that are unsupervised (c.f. [23]), though the methods are deployed from a stationary sensor. Thus, this warrants exploration into unsupervised group detection methods for mobile robots.

To address these gaps, we introduce a new method, the *Robot-Centric Group Estimation Model* (RoboGEM), which enables robots to detect human groups in real-world environments from an egocentric perspective using unsupervised learning. RoboGEM works by first estimating human velocity using dense optical flow vectors. Next, it estimates pairwise proximity between people using a pedestrian detector. Then, it combines these models

Authors' addresses: Angelique Taylor, University of California San Diego, 9500 Gilman Dr. San Diego, CA, 92093, USA, amt062@eng.ucsd.edu; Darren M. Chan, University of California San Diego, 9500 Gilman Dr. San Diego, CA, 92093, USA, dcc012@eng.ucsd.edu; Laurel D. Riek, University of California San Diego, 9500 Gilman Dr. San Diego, CA, 92093, USA, lriek@eng.ucsd.edu.

Fig. 1. This paper introduces RoboGEM, an ego-centric, unsupervised, group perception method. This figure demonstrates RoboGEM detecting groups while both the robot and pedestrians in the environment are in motion.

to compute features using joint proximity and motion predictions. Finally, it uses these features to perform hierarchical clustering analysis to detect groups.

Our approach is beneficial in several ways. First, it can be used in group tracking pipelines to improve their performance. Unlike previous approaches in the literature that only work in controlled environments and employ stationary cameras, our method is designed for ego-centric, RGB-D perception, encompassing the challenging problem of moving people from a mobile platform [65, 85, 137]. We collected an RGB-D dataset that was recorded in a crowded, sunny, outdoor environment which caused many computer vision challenges such as occlusion, shadow, and varying lighting illuminations. Thus, our approach was designed for real-world situations, using real-world data; therefore, robots using our method will well-equipped to handle everyday challenges.

The main contributions of our work are threefold. First, we present RoboGEM, a novel unsupervised, ego-centric group detection algorithm. It is straightforward to implement and can be used across many human-robot teaming scenarios. Second, our evaluation addresses an important problem in robotics, which is solving vision tasks while both the robot and pedestrians are in motion, across a challenging, real-world dataset. Third, we show that RoboGEM outperforms two top-performing algorithms by up to 30% in terms of accuracy (See Section 4.4). This work addresses an imperative need in HRI, which will contribute to our goal of enabling robots to seamlessly integrate themselves in human-centered environments. Furthermore, our work will be helpful to the robotics community as it will promote further exploration of human-robot teamwork in real-world settings.

2 RELATED WORK

In order to perform human group detection, we must first be able to detect pedestrians. Thus, we first highlight trends in pedestrian detection and discuss how those methods led to advances in egocentric perception. Then, we discuss the current state-of-the-art group-related problems in the literature which include: group dynamics, spatial behavior, and group detection (i.e., group prediction) and tracking (i.e., group identification) in HRI, as these are closely related to our work.

2.1 Pedestrian Detection

The objective of pedestrian detection is to locate people in images or consecutive frames in video streams, where the output is typically represented in the form of bounding boxes. Pedestrian detection is an important area of research for robotics and computer vision, with applications in surveillance, activity recognition, and video analysis [31, 36, 145, 145].

Past research on pedestrian detection includes approaches using either stationary or mobile sensors. Pedestrian detection from stationary cameras are typically placed overhead, where people are monitored over time [138]. Before depth sensors became prevalent in robotics, RGB images were predominantly used to detect people.

Most prior approaches use features such as Histogram of Oriented Gradients (HOG) [31] or background subtraction to narrow down the search space to find people in images [100]. Deformable part models is another popular technique that uses HOG to detect different parts of the body and is used to train classifiers for pedestrian detection [38]. Also, due to the complex geometry of the human body, other features such as color, shape, and motion are used as well. In the final stage of detection, a classifier, such as a Support Vector Machine (SVM), is used to make a decision about whether or not the image contains one or more pedestrians [83].

In robotics, sensors in addition to traditional RGB monocular cameras are often used to detect people. For example, Arras et al. [7] used a laser range finder to detect and track people's legs using a Kalman-Filter based multi-target tracking system. Jafari et al. [63] used head-mounted RGB-D sensors in a multi-hypothesis tracking system to detect and track upper bodies. They used a normalized depth-based template approach to detect upper bodies and *groundHOG* to detect people from far ranges [124]. Spinello et al. [122] detected full bodies in RGB-D data using a HOG-inspired feature, Histogram of Oriented Depths.

Recently, researchers have become interested in pedestrian detection from an egocentric perspective [14, 94, 112, 113]. Egocentric vision aims to solve perception problems from the first-person viewpoint. Applications include activity recognition, video summarization, and mapping [13, 15, 97, 112, 113]. Data is usually collected from head-mounted sensors or cameras that are mounted on mobile platforms.

Many of the aforementioned techniques have also been used for egocentric vision. However, because both the sensor and people are in motion, it is challenging to determine whether a pedestrian's change in position is a result of the sensor's motion or people moving in the environment. Thus, a motion model that describes the robot's own movements is typically used to differentiate robot from human motion [15, 94].

Egocentric vision is particularly important as robots begin entering complex human environments. It is not feasible for robots to depend on overhead cameras as a means of monitoring the robot's environment or as an extension to its vision system. Additionally, robots will encounter new situations and must be able to handle conditions with no *a priori* knowledge; therefore, they require unsupervised methods so that they can learn solely on input data.

In recent years, the computer vision community has transitioned from using hand-crafted appearance features (e.g. HOG, Haar, LBP) to using deep learning (DL) architectures to generate features for pedestrian detection. These DL architectures use Convolutional Neural Networks (CNNs) which typically consists of passing images through a series of filters such as convolution, non-linear, pooling (downsample or max pooling), rectified linear units or ReLU (normalization), and fully connected layers. The fully connected layer generates a fixed length vector that is used for classification. Using different configurations of such layers has enabled researchers to improve pedestrian detection accuracy beyond what hand-crafted featured-based approaches have been able to achieve [32, 33, 71].

There have been many approaches proposed for pedestrian detection using DL which aim to address one or more of the following challenges: using varying input image sizes [56], using region proposals effectively [49, 50], training on a full image versus training on object proposals [3, 6, 18, 37], and improving training and testing time without sacrificing accuracy [49, 103–106]. Some of the most popular methods include YOLO in its

variants [103–105], Faster RCNN [106], Spatial Pyamid Pooling networks (SSPNets) [56] and more. We use YOLO [103–105] in our work because it achieves state-of-the-art performance and runs in real-time.

2.2 Groups in HRI

Modeling group dynamics is important for robots as they work in teams. Some problems in modeling group dynamics include team decision making and synchrony. For instance, some researchers explored how robots can help mitigate conflicts in teams and how a robot's gaze influences its teammates' perception of decision making [65, 110]. Additionally, Correia et al. [29] explored how human group members in a two-human, two-robot group generate their membership preferences to these robots based on the robot's behaviors.

Synchrony is used as a way to characterize groups in order to help robots coordinate their actions with a team. For instance, Iqbal et al. [59–62] designed algorithms that model the high level actions of a human group, and measured the degree of synchrony in the group to enable a robot to coordinate its actions with the group. Additionally, Lorenz et al. [85] conducted a study on movement coordination in human-robot teams, and found that humans unintentionally coordinated their movements with robots.

Although this work serves as an important step in modeling group dynamics, they are also conducted in well-controlled environments. This can be problematic as group behavior can be unpredictable; therefore, these approaches may not be generalizable to real-world environments.

Another approach in the literature models group spatial behavior using two major constructs: proxemics and F-Formations. Proxemics is the study of human use of space during face-to-face interactions [52]. It encompasses one's personal preferences for spatial comfort zones which range from intimate to public space. This is influenced by people's culture, age, and gender backgrounds [10, 52, 89, 92, 126]. F-Formations are a systematic way of defining groups based on their sustained spatial and orientational relationship [69]. They describe how groups self-organize themselves into three spaces: (1) o-space is the center of the group, (2) p-space is where the group members stand, and (3) r-space is the space immediately outside of the group.

Much of the work done on HRI in groups has explored how a robot's behavior impacts a group's spatial behavior. For example, Vazquez et al. [135] investigated how a robot's role during a game impacts its human group partner's spatial behaviors. Vroon et al. [137] designed a reactive system that generates hypotheses for social positioning of approach, retreat, and converse behaviors while solving a group task.

2.3 Group Spatial Behaviors

Another trending topic is the exploration of spatial behaviors in public settings for robot tour guides [46, 117]. For example, Fiore et al. [40] designed a robot that actively reacts to a group's motion by performing stop and wait behaviors based on the group's needs and the urgency of the current task. Karreman et al. [68] investigated how a tour guide robot's orientation influenced visitors' orientation. However, rather than determining which people are within a group together, these robots are reactive to people in the robot's environment.

There has also been work done that uses F-Formations to model groups. The goal of F-Formation detection is to estimate the *o-space* of the group, which is the space in front of group members or in the center of the group [69]. Some work explored rotating the robot's orientation and using motion models to determine how they impact F-Formations [69, 73, 140]. Vazquez et al. [136] designed an F-Formation detection system that uses position and head orientation to track the direction of people's lower body which generates soft group assignments to track body orientation.

While this prior work aims to model groups using F-Formations, it is challenging to detect such groups from a mobile platform. Hence, current methods model F-Formations as free-standing groups, which can potentially fail in cases when people are moving.

2.4 Group Detection and Tracking

The final theme involves detection and tracking of groups and crowds [16, 23, 84, 132]. Prior work in surveillance and robotics have explored detecting moving groups in large crowds. This problem is addressed from an exo-centric perspective where the sensor is placed overhead. Many approaches include probabilistic methods such as particle filters [2, 11, 12, 21, 42, 48, 93, 99, 101, 141–144], graph-based approaches such as generalized minimum clique graphs [19, 39, 70, 81, 95, 146], clustering-based methods such as k-means and agglomerative clustering [45, 47, 120, 121], as well as methods based on the social force model [78, 88, 119, 123].

Group detection and tracking has also been explored from an ego-centric perspective using Multiple Hypothesis Tracking (MHT) [16, 24, 75, 76, 86, 91], fluid dynamics [116], and clustering [23, 129, 131]. MHT is the most popular method which formulates group tracking as a combinational selection problem where a set of hypotheses from the previous and current iteration of the algorithm are evaluated in order to perform data association. Due to the computational complexity of MHT, many approaches select the k-best hypotheses [75, 76, 91]. Also, researchers have addressed group splitting, merging, and size estimation using MHT [76, 86].

The probabilistic approach proposed by Choi et al. [24] localizes and classifies *structural groups* in a single image to encode interactional features between people in groups using bottom-up interaction potentials, intragroup potentials, and background potentials. The work done by Choi et al. relies on several features such as individual poses (standing, sitting on an object, sitting on the floor), and 8 different viewpoints (front, front-left, back-right, etc.). However, such features are not readily accessible for mobile robots and gaining access to these features would greatly increase an algorithm's complexity. For example, this method requires real-time activity recognition (to detect individual poses) and a multi-sensor network (for multiple viewpoints). Also, the work done in [24] does not run on sequential video data which is typically the case for perception problems on mobile robots.

Brščić et al. [16] designed a probabilistic model of spatial formations of pedestrians to predict two and three person groups. However, in real-world settings, robots can encounter groups exceeding a membership of three, which could lead to challenges when the robot is required to work alongside such groups.

Clustering-based methods typically consist of estimating features that characterize groups and then find clusters within these features to detect groups. For example, Chatterjee and Steinfeld [23] estimated dense crowds by finding clusters in 3D point clouds. They used these clusters to predict moving pedestrians in crowds.

The work done in the Spencer project has by far made the most headway in detecting and tracking groups from an ego-centric perspective. The project's goal is to design an assistive robotic platform that guides travelers through busy airports [132]. As a part of this project, Linder et al. published an evaluation framework that detects and tracks human groups [84]. Although group tracking methods are arguably more useful in practical situations than group detectors, they depend on group detectors for accurate tracking performance. Linder et al. [84] identified the pedestrian detector as a key component of the tracking pipeline which requires further attention in the research community to improve tracking accuracy.

We have identified several gaps in the literature which we plan to address in our work. First, there are many approaches that are conducted in controlled, predictable environments. This can hinder computational models as they do not generalize to real-world environments. Thus, we collect our data in a naturalistic setting, capturing real people in the real-world.

Additionally, unlike methods that make an underlying assumption that people in the environment are a part of a group, we aim to detect different groups throughout the robot's environment including people from near and far distances. Thus, in situations where robots are working with a team, they will have the ability to detect their team members as they move throughout the environment.

Although F-Formations are important for free-standing groups, current approaches have an underlying limitation as it is challenging to estimate the o-space as the group moves in the environment. Additionally, most prior work use exo-centric (i.e. birds-eye view) sensors which leads to infeasible sensing systems in everyday

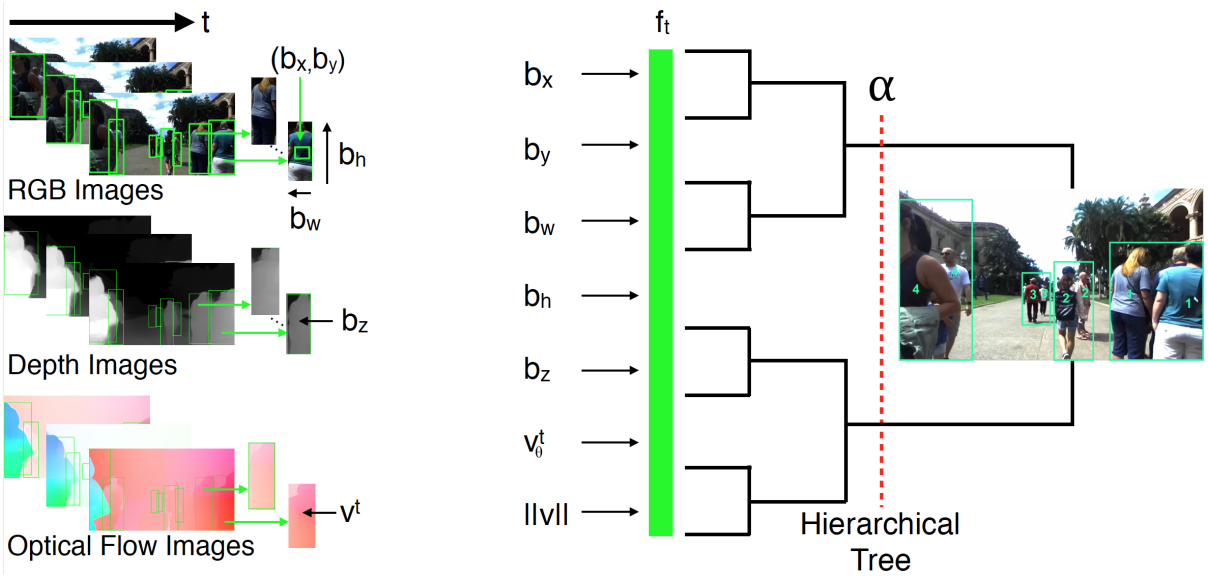

Fig. 2. Overview of RoboGEM. First, RoboGEM detects pedestrians in RGB images using an off-the-shelf detector. This provides the bounding boxes (BB) $\langle b_x^{n,t}, b_y^{n,t}, b_w^{n,t}, b_h^{n,t} \rangle$ for pedestrians in the data, where $b_x^{n,t}$ and $b_y^{n,t}$ are the BB centroids, $b_w^{n,t}$ is the width, and $b_h^{n,t}$ is the height. In parallel, the pedestrian motion estimation module computes optical flow vectors $v^t = \langle v_x^t, v_y^t \rangle$ from t to $t+1$. RoboGEM uses the magnitude $||v||$ and orientation v_θ^t. Next, the group detection module computes the mean of the depth pixels for the BB, $b_z^{n,t}$ for each person. Then, RoboGEM concatenates the aforementioned features into a vector $f^{n,t}$. We use these features to perform hierarchical clustering to detect human groups.

environments. Alternatively, we use joint proximity and motion estimations to detect groups, as this is most feasible in real-world environments to date. Also, we use egocentric perspective sensors so that robots are not required to depend on external monitoring systems.

Finally, the method proposed by Linder et al. [84] is most comparable to ours and addresses some of the same challenges. However, its main drawback is that it uses supervised learning and requires large manually labeled datasets for training. To address this gap, we designed an unsupervised algorithm that detects human groups. To our knowledge, this paper is the first to address unsupervised detection of human groups from the egocentric perspective of a mobile robot.

3 ROBOT-CENTRIC GROUP ESTIMATION MODEL

The goal of RoboGEM is to enable robots to detect human groups from an egocentric perspective. We use the definition of groups from Linder et al., [83] which states that groups are two or more people in close proximity to one another with a common motion goal. Our method is comprised of three modules: pedestrian detection module P, pedestrian motion estimation module V, and group detection module G (See Figure 2).

RoboGEM can be used with any standard RGB-D sensor or stereo camera as long as they provide calibrated RGB and depth image pairs. Additionally, it was also designed for mobile robots, so it can detect groups under stationary and mobile sensor motion scenarios. Furthermore, it does not require *a priori* knowledge of the robot's environment (e.g. indoor/outdoor, objects present, etc).

ALGORITHM 1: *Group_Detection*($f^{n,t}, \alpha$)
Assigns group identifiers to pedestrians in RGB-D data.

Input : f is a list containing features.
α is the inconsistency coefficient.
Output: C is a list containing cluster identifiers for each pedestrian.
$dist_t$ = {} // matrix of pairwise euclidean distances between observations.
Z = {} // matrix containing hierarchical clustering tree.
for t = 1 *to* T **do**
  $dist_t \leftarrow L2NORM(f^{n,t})$
  $Z \leftarrow LINKAGE(dist_t)$
  $C \leftarrow CLUSTER(Z, \alpha)$
end
Return C

L2NORM($f^{n,t}$): Returns the euclidean distance between each feature in $f_{n,t}$ (See Eq. 3).
LINKAGE($dist_t$): Returns a similarity measure between clusters in $dist_t$ using average linkage (See Eq. 4).
CLUSTER(Z): Returns group cluster identifiers for each person observed in in an image.

Our approach leverages spatiotemporal observations of people to cluster them into groups. The overall intuition of our approach is that groups tend to walk in similar directions, with similar motion patterns, and in close proximity to each other [84]. Using calibrated RGB-D images, RoboGEM identifies people, measures their proximity, and determines their velocity. These features which are then used by the group detection module, which employs agglomerative hierarchical clustering to detect groups.

We make the following assumptions: the robot is mobile and moving around in a space where $n = 1, 2, 3, \ldots, N$ is the number of people present at time (or frame) t, where $t = \{1, 2, 3, \ldots, T\}$ for T-number of frames in a video sequence. P generates bounding boxes (BB) at time t, $b^{n,t} = \langle b_x^{n,t}, b_y^{n,t}, b_w^{n,t}, b_h^{n,t} \rangle$ which is the centroid column and row, width, and height respectively.

V estimates velocity $v^t = \langle v_x^t, v_y^t \rangle$, which is a vector containing motion estimates from t to $t + 1$. Let the flow vectors for $b^{n,t}$ be denoted $v^t(b^{n,t})$. A local feature vector for an image at t for a pedestrian n is denoted $f^{n,t} = \langle b_x^{n,t}, b_y^{n,t}, b_w^{n,t}, b_h^{n,t}, b_z^{n,t}, ||v^t||, v_\theta^t \rangle$, where the BB coordinates are normalized between 0 and 1, as explained in Section 3.3. The output of RoboGEM is $C^{n,t} \in \mathbb{R}^N$ vector which holds a group number or cluster *identifiers* for $b^{n,t}$.

3.1 Pedestrian Detector Module

We use an off-the-shelf pedestrian detector (YOLO) that has state-of-the-art object detection performance (81.3% average precision) with a reasonable frame rate (40-90 frames per second on a GPU) [103, 104]. Although RoboGEM itself is unsupervised, YOLO is deep learning-based. However, we did not train YOLO, as its pre-trained model produced sufficient results. YOLO divides images in a grid which generates a class probability map and bounding boxes with class confidences. Then, it performs regression on these data to infer the bounding box coordinates.

However, during algorithm development, we found that P performs poorly with people at far distances. For example, when people are far away from the robot, P generates a pedestrian patch that covers a wide field-of-view around many people. Therefore, we preprocessed the pedestrian detection instances by excluding any bounding boxes with a width $\geq w \times (0.75)$ where w is the total image width.

3.2 Pedestrian Motion Estimation Module

In the next step of RoboGEM, V estimates pedestrian motion using optical flow. Optical flow is ideal for group detection tasks, as it can provide a quantitative measure of pedestrian velocity [43].

Using this feature, we group people that walk in similar directions, including moving and stationary pedestrians. The performance of optical flow is highly dependent on the degree of ego-motion of the sensor, as it is often subject to large amounts of noise while the sensor is in motion. Thus, we require a method which provides dense optical flow vectors while reducing noise.

We use FlowNet 2.0, a neural network-based optical flow algorithm which has been used for motion segmentation and action recognition [58]. It uses a stacked FlowNet architecture, and incorporates image warping to achieve smooth motion fields. This enables us to decrease the noise caused by sensor motion and better detect people's motion.

Given two consecutive images as input from time t to $t+1$, FlowNet 2.0 computes the partial derivative of image pixels with respect to the spatiotemporal coordinates. It generates an image representation where each pixel is a velocity vector $v^t = \langle v_x^t, v_y^t \rangle$. This enables RoboGEM to discern between people that are not walking in the same direction by computing the magnitude and orientation of these vectors.

3.3 Group Detection Module

In the final step, RoboGEM performs human group detection. It computes features using joint motion and proximity estimation. As previously mentioned, people walking in groups tend to walk in similar directions, in close proximity, and with similar motion patterns. Additionally, we observe that people walking in groups are walking at similar distances from the robot and therefore should have similar sized pedestrian BBs.

Given a sequence of spatiotemporal images, RoboGEM runs P. Then, it estimates flow vectors within $b^{n,t}$ and stores them in $v^t(b^{n,t})$. RoboGEM estimates the mean of the x and y components of the velocity vectors $v_x^t(b^{n,t})$ and $v_y^t(b^{n,t})$, denoted as μ_x and μ_y respectively. The direction of pedestrian motion is estimated using Eq. 1 and the magnitude is estimated using Eq. 2.

$$v_\theta^t \leftarrow tan^{-1}\frac{\mu_y^t}{\mu_x^t} \tag{1}$$

$$||v^t|| \leftarrow \sqrt{(\mu_x^t)^2 + (v\mu_y^t)^2} \tag{2}$$

μ_x^t is the mean x component of velocity at time t.
μ_y^t is the mean y component of velocity at time t.

We perform a similar procedure on depth images to estimate proximity from the robot to pedestrians. However in this case, we must consider which pixel values correspond to the distance from the pedestrian to the robot as some pixels are from the background; therefore, we use the mean of the pixel values in $b^{n,t}$ as a distance measure. In order to delineate between people that are close to the robot from those that are far away, we use the width and height of the BB which are $b_w^{n,t}$ and $b_h^{n,t}$ respectively.

The final feature is the proximity between people on the image plane. This feature uses the raw $b_x^{n,t}$ and $b_y^{n,t}$ positions as they are the centroids of the pedestrian BB. Although this feature can perform poorly when one person walks in front of another person, this feature combined with the depth feature increases the robustness of RoboGEM. Once all features are computed, we normalize them between 0 and 1 and then concatenate all the features into a single vector $f^{n,t} = \langle f^{n,1}, f^{n,2}, \ldots, f^{n,T} \rangle$.

As the overarching goal of our work is to detect human groups using methods that require no training, RoboGEM leverages the hidden structure in data. One such method is hierarchical clustering, which discriminates a group of objects into sets of clusters of similar likeness. Hierarchical clustering is classified into one of two types: divisive and agglomerative.

Divisive clustering follows a top-down approach, by first grouping all observed objects into one cluster. Then, it iteratively divides clusters into smaller ones, until all of the objects are assigned to its own cluster, or when a termination condition is reached. Alternatively, hierarchical clustering analysis (HCA) performs the opposite operation by employing a bottom-up approach in an agglomerative fashion. Objects are first treated as separate clusters, and iteratively merged until all objects are merged until a termination condition is reached.

We employ HCA because it is most suitable for our problem, as we start with atomic units which are represented by individual pedestrian detection instances and aim to cluster them into human groups. Also, it does not require large amounts of the data, is simple to implement, and does not require that the number of groups is defined *a priori*.

RoboGEM computes the pairwise L^2-norm between observations in $f^{n,t}$ which yields a matrix $dist_t$ (See Eq. 3). It groups $dist_t$ into a binary HCA tree by linking observations with close proximity using average linkage. The distance between two clusters L, is defined in Eq. 4.

$$dist_t \leftarrow \left\| \left(f^{j,t} - f^{k,t} \right) \right\|_2 \tag{3}$$

$$L(C^{j,t}, C^{k,t}) = \frac{1}{|C^{j,t}||C^{k,t}|} \sum_{f^{j,t} \in C^{j,t}} \sum_{f^{k,t} \in C^{k,t}} dist(f^{j,t}, f^{k,t}) \tag{4}$$

where $j \neq k$
$f^{j,t}$ is a feature vector for person j.
$f^{k,t}$ is a feature vector for person k.
$C^{j,t}$ is cluster j.
$C^{k,t}$ cluster k.

Then, it prunes the hierarchical tree to partition the observations into clusters. There are two methods for pruning which include identifying the number of maximum clusters (similar to k-means) or by finding natural divisions in the observations.

Due to the challenges of predicting the amount of people entering and leaving the robot's field-of-view, we find natural divisions in the data using an *inconsistency coefficient*, α which is a threshold for each link in the hierarchical tree. It compares the height of the clusters represented in a tree with the average height in a level within the tree. Therefore, a high α corresponds to dissimilar observations and a low α corresponds to more similar observations. In order to choose an α that provides the best accuracy of groups, we conducted a pilot experiment using simulated data and experimented with alpha values ranging from 0 to 4 in increments of 0.1. We found that $\alpha = 0.1$ had the best results while $\alpha \geq 1$ had the worst results. Thus, we report our findings using $\alpha = 0.1$.

Pruning the hierarchical clustering tree provides a vector $C^{n,t} \in \mathbb{R}^N$ which holds a group number or cluster *identifiers* for $b^{n,t}$. Then, we perform pruning on the clusters to detect groups. For example, suppose $C^{n,t} = \langle 1, 2, 2, 3 \rangle$. Pedestrian at index 0 has a group identifier of 1. Pedestrian at index 1 has a group identifier of 2. Pedestrian at index 2 has a group identifier of 2 and the pedestrian at index 3 has a group identifier of 3. Therefore, pedestrian at index 1 and 2 are in a group because they have the same group identifier. In this case, we must remove clusters with a $frequency < 2$; therefore, the resulting $C^{n,t} = \langle 2, 2 \rangle$. To accomplish this, we use the frequency of unique identifiers in $C^{n,t}$ and remove pedestrians from $C^{n,t}$ that have identifier with a $frequency < 2$ as groups are two or more people. Then, we compute the merged BB of groups, denoted as M. The final representation of groups includes the groups' BB and their group identifier.

4 EXPERIMENTS

4.1 Data Collection

To evaluate RoboGEM, we required a pedestrian dataset consisting of spatiotemporal RGB-D images captured from an egocentric viewpoint of a mobile robot. Furthermore, we were interested in evaluating our algorithm on data that consisted of candid human groups. While many ego-centric pedestrian datasets exist (c.f. Caltech [36], INRIA [31], Daimler [41]), they do not work for our intended purpose because they do not simultaneously contain depth information and egocentric motion. Moreover, previous datasets are often captured in spaces where social gatherings of pedestrians are sparse. Also, other datasets such as UT Interaction [111], Collective Activity [26], Collective Activity Extended [34], Volleyball Activity [57], Nursing Home [35] are not adequate for our evaluation because they do not contain group annotations. There are many publicly group detection datasets (i.e., Crowds-By-Example [80] and BIWI Walking pedestrians [98]); however, these are datasets captured from a stationary, exo-centric perspective which is not representative of robot vision, they do not contain depth data, and the pedestrians are represented as a point instead of a bounding box as done in RoboGEM. To the best of our knowledge, there is one publicly available egocentric group detection dataset, Structural Groups [24]; however, it does not contain depth data and spatiotemporal observations of groups.

Instead, we sought to evaluate RoboGEM on a dataset that encompasses real-world challenges which robots might encounter when operating in public, crowded spaces. Thus, we acquired a challenging real-world dataset that was captured "in-the-wild"; therefore, we had no control over what people did or how they behaved, which is important for robots working in everyday settings. Some challenges which appear in our dataset include: variable lighting, occlusion, chaotic motion trajectories, and motion blur. These challenges provide a useful benchmark, because a robot might be expected to work in both indoor and outdoor environments, where lighting conditions can change dynamically as the robot navigates from one position to another. Furthermore, a robot's vision can be suddenly occluded or blurred, where the robot needs to estimate the state of its surroundings. Most importantly, we are interested in challenges involving a mobile robot navigating around groups of pedestrians, which are highly unpredictable.

Thus, we collected our own RGB-D pedestrian dataset. We mounted a ZED stereo camera at human height on a Double Telepresence Robot (See Fig. 1). The ZED was configured to capture video at approximately 20 frames per second, at a resolution of 640×360. The robot collected data while being teleoperated using the Double mobile application. Consistent with other popular robot vision datasets [5, 74], we acquired our data in discontinuous segments to capture a wide range of real-world conditions. The collection site consisted of a large, outdoor public park across several different locations to diversify lighting conditions, degree of crowdedness, and to capture varied motion patterns. The robot roamed around the park moving through corridors, on sidewalks, and through large crowds during the daytime where people were observed walking, eating, and viewing local nearby landmarks.

In total, our dataset consists of 16,827 RGB-D images, representing 1.5 hours of video. The total number of groups between frames in our dataset is 5,423 (not unique groups). A member of our team labeled 14,710 images with bounding boxes around groups. In order to detect groups, we adopt the definition of groups used by Linder et al. [83]. This definition states that groups are two or more people in close proximity to each other with a common motion goal.

In order to validate our labels, a second member of our team labeled 2,000 randomly selected images from our dataset. We employed a validation method in concert with other leading methods in the field (e.g. the COCO dataset [82]). We computed the precision and recall of both team members' labels which is comparable to COCO's expert [82]. The precision is 0.83 and the recall is 0.79 at an Intersection-over-Union (IoU) of 0.4, where IoU is a ratio that measures the overlap of a predicted box and a groundtruth box (See Section 4.2 for a detailed discussion).

4.2 Experimental Setup

We evaluate RoboGEM by comparing it to three methods: (1) an extension of RoboGEM(RoboGEM:HC+LDA-L1, (2) Spencer's group detector (Spencer:SVM) [83], and (3) the group detection method developed by Solera et al. (Solera:G-MITRE) [121].

Experiments were conducted on a Dell Inspiron Intel Core i7 laptop, with 16 GB of RAM, 1 TB HDD, with a NVIDIA GeForce GTX960M GPU. The machine ran Ubuntu 14.04 Linux and all algorithm development was completed in MATLAB.

We performed two steps to ensure a consistent evaluation among all methods. First, we generated pedestrian bounding boxes using the same pedestrian detector for all methods, YOLO [104, 105]. Second, we merged the pedestrian bounding boxes to form group bounding boxes. Then, we followed the evaluation protocol described in Section 4.3.

4.2.1 Comparison to RoboGEM extension (RoboGEM:HC+LDA-L1). To create the first comparator method, we extend upon RoboGEM, by using a post-processing method, Linear Discriminant Analysis based on L1-norm maximization (LDA-L1). We choose this method for comparison because it minimizes the within-cluster dispersion of groups, while also maximizing the between-cluster dispersion of groups to improve clustering performance; therefore, in situations when RoboGEM did not perform well, we expect LDA-L1 to improve its performance.

First, RoboGEM is used to estimate groups, then LDA-L1 is applied. This yields local optimal projection vectors for groups [147]. The goal of traditional LDA is to reduce feature dimensionality by learning a set of projection vectors $W = [w_1, w_2, \ldots, w_n] \in \mathbb{R}^k (k < N)$ that constitute a low-dimensionality linear subspace. Unlike traditional LDA, LDA-L1 is more robust to outliers and uses a greedy search method to obtain $N - 1$ local optimal projection vectors. To summarize, we address the following optimization problem using notation that is consistent with [147]:

$$\boldsymbol{w}^* = \arg\max_{w} F(\boldsymbol{w}) \text{ subject to } w^T w = 1 \tag{5}$$

where $F(\boldsymbol{w})$ is the objective function as follows:

$$F(\boldsymbol{w}) = \frac{\sum_{i=1}^{C} N_i \left\| \boldsymbol{w}^T (\overline{x}_i - \overline{x}) \right\|_1}{\sum_{i=1}^{C} \sum_{j \in C_i} \left\| \boldsymbol{w}^T (x_j - \overline{x}_i) \right\|_1} \tag{6}$$

$\boldsymbol{w}^*$ is one projection vector which is the local optimal solution of F.
$\mathbf{w}$ is the columns of W.
$\overline{x}_i$ is the i^{th} group mean vector.
$\overline{x}$ is the global mean (i.e. mean of all features) vector of f^t.
x_j is the sample vector projected on W.
C is the number of groups.
C_i is the i^{th} member of a group.

We use W to project the features f^t onto a linear subspace and find the hierarchical clusters of these projected features. We refer readers to Zhong and Zhang's work [147] for a detailed description of the LDA-L1 method and for a proof of convergence.

4.2.2 Comparison to Spencer Group Detector (Spencer:SVM). Second, we compared RoboGEM to the state-of-the-art supervised group detection method used in the Spencer project [8, 84, 86, 132]. The system pipeline of this approach includes pedestrian detection, group detection, and group tracking. Although the Spencer pipeline includes group tracking functionality, to facilitate a fair comparison we only used its group detection method in our evaluation. This represents an important test, however, as the detector's performance greatly impacts

how well the group tracker performs. Additionally, to our knowledge, this is the most comparable method in the literature, and shares our definition of groups (a collective of individuals in close proximity with a common motion goal [83]).

This approach uses RGB-D data and 2D laser scanner data to compute velocity, euclidean distance, and orientation features. In contrast to our method, this approach uses supervised learning to detect groups. In addition, our dataset did not contain 2D laser scans; therefore, to make a fair comparison to RoboGEM, we re-implemented Spencer to accept depth images. We used the same pedestrian detector that is used in RoboGEM to detect pedestrians [103]. Then, we use optical flow to estimate velocity and orientation. Following the approach used by Linder et al. [83], we compute the pairwise difference in bounding box x position, bounding box y position, velocity, and orientation between all pedestrians. Then, we trained an SVM to learn a pairwise social relation between all pedestrians on our dataset using the aforementioned features. This generates a social relation score between 0 and 1. Then, we constructed a social network graph where each node is a pedestrian and the edges are weighted by the pedestrians social relation score. Similar to [83], we disregard edges with a score less then 0.5. Finally, we compute connected components in the graph to detect groups using the Depth First Search algorithm.

4.2.3 Comparison to Solera (Solera:G-MITRE). Finally, we compared RoboGEM to the group detection method by Solera et al. [121]. This method employs correlation clustering and a structured SVM. They propose a loss function that models clustering constraints for crowds of people. Because this method models large crowds, it becomes computationally expensive; therefore, the authors use a spanning tree representation in their G-MITRE loss function and then find the connected components in these graphs to detect groups.

To evaluate Solera on our dataset, we needed to transform our data into a format compatible with their codebase[1]. The method requires a unique pedestrian ID for all people in our dataset, so we first ran a pedestrian detector (YOLO) to generate pedestrian bounding boxes. Then, we ran a pedestrian tracker [139] to generate pedestrian IDs. Next, we generated a clusters file where each line contains the pedestrian IDs for a group. Also, we generated a trajetories file that contains the frameID, pedestrian ID, x, and y coordinates indicating where each pedestrian is located in a image. Finally, we converted our bounding boxes to a single point on the ground plane, which is located at the bottom center of the bounding boxes.

To keep our data consistent with the dataset used by Solera et al. (Crowds-By-Example) [80], and match their system's expectations, we normalized the x and y coordinates in our dataset to match the scale of the x and y coordinates in their dataset.

We inputted the cluster and trajectory files into Solera, which generated group clusters where each group is represented by a set of pedestrian IDs. [2]

4.3 Evaluation Metrics

We measure group detection performance using three metrics: (1) accuracy versus Intersection-over-Union (IoU), (2) log-average miss rate versus false positives per image (FPPI), and (3) accuracy versus depth threshold [20, 103, 106]. We use these metrics because they evaluate our method's accuracy and how the accuracy is affected by the detection distance range. For example, pedestrians that are far away from the robot have small bounding boxes which have high false positive rates. Therefore, we conducted experiments to investigate the impact of detection range on group detection performance.

[1] http://imagelab.ing.unimore.it/group-detection/

[2] One aspect of the code that we modified was the maximum number of iterations for convergence which was originally set to 300; however, we changed this parameter to 700 because we found that any number of iterations less than 700 did not generate groups.

Method	Accuracy ↑ (IoU = 0.4, Depth = 0.0)	Accuracy ↑ (IoU = 0.4, Depth = 0.4)	Precision ↑ (IoU = 0.4, Depth = 0.0)	Recall ↑ (IoU = 0.4, Depth = 0.0)
Spencer:SVM	0.27	0.40	**0.34**	0.16
Solera:G-MITRE	0.07	0.07	0.11	0.04
RoboGEM:HC+LDA-L1	0.36	**0.45**	0.33	**0.24**
RoboGEM:HC	**0.37**	**0.45**	**0.34**	**0.24**

Table 1. The results with an IoU threshold of 0.4 on our dataset. We measure accuracy at depth thresholds of 0 and 0.4. Precision and recall are measured with a depth threshold of 0, which includes all pedestrians, regardless of distance and is thus most reflective of our algorithm's overall clustering performance. For all metrics, higher is better.

4.3.1 Accuracy vs. Intersection of Union. In the first experiment, IoU measures how closely RoboGEM matches the ground truth, or how well it was able to match the performance of human annotators (See Eq. 7) [36].

$$\frac{B \cap GT}{B \cup GT} \geq \delta \qquad 0 \leq \delta \leq 1 \tag{7}$$

B is a rectangular bounding box predicted by a detector.
GT is a ground truth rectangular bounding box.
δ is the IoU threshold.

We predict paired BB from GT to B using the Jonker-Volgenant algorithm [64]. This is a greedy algorithm that is used to search for the best matching pair of ground truth and experimental BB which yield the highest IoU score.

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \tag{8}$$

ACC is the accuracy.
TP is the total number of true positives.
TN is the total number of true negatives.
FP is the total number of false positives.
FN is the total number of false negatives.

This enables us to apply a threshold δ to IoU in order to ensure a fair assessment of the overlap between the predicted and ground truth BB. A true positive, TP, corresponds to a detection with an IoU that does not exceed δ. Otherwise, the detection is considered a false positive, FP.

When the ground truth does not contain a bounding box and RoboGEM does not detect a group in the same area, this represents a true negative, TN. IoU is normalized between values of 0 and 1, and is characterized by two extremes. For instance, an IoU value of 0 equates to zero percent overlap between the algorithm's predicted bounding box and the ground truth bounding box. In the opposite extreme, an IoU value of 1 equates to perfect overlap between the algorithm's predicted box, and that of the ground truth. In order to evaluate a correct detection we use the IoU value as a threshold. For example, if the IoU is set to 1, only predicted boxes that have perfect overlap with the ground truth is considered a correct detection. However, in the context of pedestrian detection, people are often treated as non-rigid or deformable objects, where an IoU ≥ 0.4 is the standard value for a predicted box to be considered a correct detection [36].

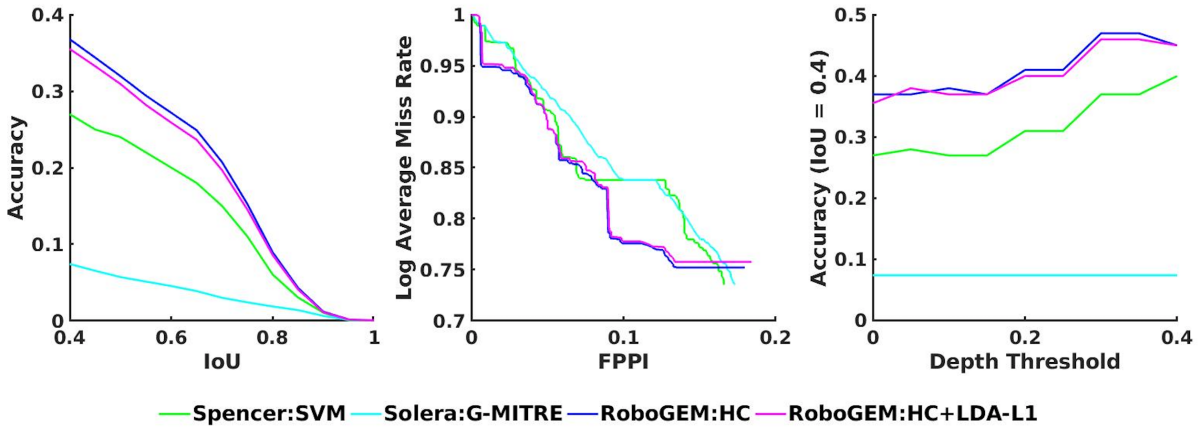

Fig. 3. Results of Spencer:SVM, Solera:G-MITRE, RoboGEM:HC, and RoboGEM:HC+LDA-L1: Accuracy vs. IoU (higher is better), Log-Average Miss Rate vs. FPPI (lower is better), and Accuracy at Fixed IoU = 4 vs. Depth Threshold (higher is better).

4.3.2 Log-Average Miss Rate vs. False Positives-Per-Image. The second metric that we use in our evaluation is log-average miss rate versus false positives per image (FPPI). This metric is similar to average precision, but is a more stable and informative assessment of performance [36]. It is computed by averaging the miss rate at nine FPPI rates evenly spaced in log space [36]. Hence, lower curves indicate better performance.

4.3.3 Accuracy vs. Depth Threshold. The third evaluation metric is accuracy versus depth threshold. As pedestrians move further from the robot, it becomes challenging to detect them. For instance, Linder et al. [83] ignored detection instances greater than 12 meters due to annotation challenges, extreme occlusions, and increased inaccuracy of sensor calibration. Therefore, we conducted experiments to evaluate the accuracy of RoboGEM at various depth thresholds for a fixed IoU of 0.4, or the standard criterion used in pedestrian detection.

We normalized the depth maps between 0 and 1 on a per-image basis and applied depth thresholds from 0 to 0.40 (about 8 meters) in 0.05 increments to the depth map. This gives us a comparable distance threshold to Linder et al. [83] at about 10 meters (the ZED depth range is from 0.5 - 20 meters). We use these depth thresholds because they range from using all of the bounding box data to a strict selection of pedestrians which are close to the robot. If the mean of the depth map within $b_z \geq threshold$, we consider these pedestrian detection instances in our algorithm; otherwise, we do not consider the pedestrian detection.

4.4 Results

Figure 3 presents our results, which compares the three comparator methods against RoboGEM. Figure 3 (left) shows accuracy at various IoU thresholds within the range of 0.4 and 1, where higher is better. As expected, the performance of all the methods degrade as IoU increases, because the overlap criterion becomes more strict. Our results show that RoboGEM outperforms Spencer:SVM [8, 84, 86] and RoboGEM:HC+LDA-L1 in detection accuracy by 10% and 1%, respectively. Also, Spencer:SVM outperforms Solera:G-MITRE by 20%. This suggests that both versions of RoboGEM are superior to both Spencer [84] and Solera [121].

Figure 3 (center) shows results for log-average miss rate with varying FPPI where lower is better. RoboGEM:HC and RoboGEM:HC+LDA-L1 have similar miss rate performance, and they both outperform Spencer:SVM. RoboGEM:HC, RoboGEM:HC+LDA-L1, and Spencer:SVM have comparable precision. This indicates that there is a high false positive rate, consistent with the findings in Linder et al. [84]. RoboGEM:HC and RoboGEM:HC+LDA-L1

have similar recall performance, both outperforming Spencer:SVM and Solera:G-MITRE. This suggests that RoboGEM is 50% better at recalling groups when compared to Spencer:SVM and Solera.

Figure 3 (right) shows accuracy results at a fixed IoU threshold of 0.4, with varying depth thresholds from 0 to 0.4 in increments of 0.05 where higher numbers are better. In general, as the depth threshold increases to 0.4 (about 10 meters), the performance of all methods improve; RoboGEM:HC and RoboGEM:HC+LDA-L1 outperforms Spencer:SVM and Solera:G-MITRE. Additionally, Solera:G-MITRE does not rely on depth data. As a result, its performance is constant at an accuracy of 0.07 while RoboGEM:HC, RoboGEM:HC+LDA1, and Spencer:SVM accuracies increase as the depth threshold increases.

5 DISCUSSION

In this paper, we introduced RoboGEM, an unsupervised robot-centric human group detection algorithm. Our method outperformed two state-of-the-art supervised methods, Spencer [83], by 10% and Solera [121] by 30%. Furthermore, RoboGEM is 50% better at recalling groups than Spencer, and substantially better at recalling groups than Solera. All of the methods presented have a low precision rate (high false positive rate), which is consistent with the findings presented by Linder et al. [84]. Therefore, RoboGEM has a comparable precision rate to the state-of-the-art. Also, as the depth threshold increases to 0.4, our method performed with an overall higher accuracy.

In contrast to related work, which use fixed sensors and/or supervised learning, our work explores how unsupervised methods can be used to address challenging problems in noisy environments. Although our implementation used stereo images, our approach is generalizable to other RGB-D sensors, as well as in indoor and outdoor settings. RoboGEM is simple to implement and was developed for robot-vision in real-world environments. Furthermore, our contributions include the following: an unsupervised group detection algorithm, an evaluation of our approach on an egocentric real-world dataset, where both pedestrians and the robot were in motion at the same time, and we showed that our method outperforms two top-performing algorithms.

Our work has several implications for the HRI community. It shows that unsupervised group detection methods have the potential to outperform supervised methods using noisy, real-world data. This work can help encourage others to investigate human-robot teaming in real-world environments rather than exploring problems in well-controlled spaces. For instance, our method can be used in healthcare settings where robots are responsible for working with clinicians to care for patients [109, 128]. This can help the robot understand how it can appropriately enter and exit team interactions.

In addition, our method can be used when a robot needs to help a group complete a collaborative task. This task might consist of coordinating with a team in which our method is used to detect where the robot's team members move over time. It can be used in conjunction with group activity recognition methods [4, 9, 25, 27, 34, 118]. Also, RoboGEM can be incorporated into existing pipelines, like the Spencer project [132], to improve group detection performance. Furthermore, our method can be used in autonomous driving systems to give them a high-level understanding of group motion on the road.

Recently, researchers have used deep learning to address vision problems such as pedestrian detection [104]. Such approaches are beneficial because they can learn on large datasets. However, training for long periods of time to achieve improved accuracy remains a challenge. As a result, there is a tradeoff between our approach, which uses hand engineered features to detect groups without training, and those that are more data-driven and require training on large datasets. As such, there are exciting opportunities to apply deep learning to the group detection problem, which may result in improved group detection.

As robots become more integrated into our daily lives, they are expected to work alongside groups of people in teams. However, when robots cannot effectively detect its team members, it can potentially cause confusion among the team and teams working in close proximity to it. As a result, this situation can negatively impact

Fig. 4. Examples of successful group detections. The green and blue boxes show the ground truth and predicted labels respectively.

a robot's performance in a team. As a result, robot group detection algorithms that perform poorly can cause people to lose trust in the robot which is an important area of research [53, 87]. Alternatively, a robot with more accurate group detection algorithms can work alongside its team more fluently because the interaction is not negatively impacted by poor robot perception. Robots can effectively detect their team members and coordinate their actions with the team to accomplish their shared goals. This can preserve trust in human-robot teams and can potentially increase trust between humans and robots.

In the future, we look forward to improving RoboGEM in order to investigate collective group motion. To approach this direction of exploration, we plan to use probabilistic reasoning to detect and track people over time. Furthermore, we will incorporate odometry data to be used in a motion model to reason about a pedestrian's movement relative to the robot's movement. We are also considering using more accurate sensors, such as LIDAR, for more precise motion estimation as our observation aligns with [84] that the depth measurements are often imprecise. For example, this alternative sensor may be better to cover farther ranges. Finally, we plan to use this extension to investigate how robots can use this knowledge to interact and work in teams with groups. Additionally, prior work has shown great potential in group activity recognition; therefore, combining group detection with group activity recognition is another exciting area of future exploration [4, 9, 25, 27, 34, 118].

Group detection is an important problem in robotics and requires further attention in order to improve group tracking performance. Also, by designing more accurate group perception methods, robots can better predict pedestrians motion intentions. This can enable robots to employ safer and more socially aware navigation in crowded environments. By incorporating egocentric vision and unsupervised learning in our algorithmic design, we hope that our method can be easily used in other robotics problems such as navigation, human-robot teaming, and coordination.

6 ACKNOWLEDGEMENTS

This material is based upon work supported by the National Science Foundation under Grant Nos. IIS-1527759 and DGE-1650112.

REFERENCES

[1] 2019. Diligent Robotics. (2019). https://diligentrobots.com/[Accessed31Aug.2019]

[2] A. Al Masum, M. H. Rafy, and S. M. Rahman. 2014. Video-based affinity group detection using trajectories of multiple subjects. In *International Conference on Electrical and Computer Engineering (ICECE)*. IEEE, 120–123.

[3] B. Alexe, T. Deselaers, and V. Ferrari. 2012. Measuring the objectness of image windows. *IEEE transactions on pattern analysis and machine intelligence* 34, 11 (2012), 2189–2202.

[4] M. R. Amer and S. Todorovic. 2011. A chains model for localizing participants of group activities in videos. In *International Conference on Computer Vision (ICCV)*. IEEE, 786–793.

(a) (b) (c)

Fig. 5. Examples of unsuccessful group detections. The green and blue boxes show the ground truth and predicted labels respectively. Images (a) and (b) show false negatives, where groups existed but went undetected by RoboGEM. Image (c) depicts a false positive, where there is not a group but RoboGEM detected one.

[5] Phil Ammirato, Patrick Poirson, Eunbyung Park, Jana Košecká, and Alexander C Berg. 2017. A dataset for developing and benchmarking active vision. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 1378–1385.

[6] P. Arbeláez, J. Pont-Tuset, J. T. Barron, F. Marques, and J. Malik. 2014. Multiscale combinatorial grouping. In *Computer vision and pattern recognition (CVPR)*. 328–335.

[7] K. O. Arras, S. Grzonka, M. Luber, and W. Burgard. 2008. Efficient people tracking in laser range data using a multi-hypothesis leg-tracker with adaptive occlusion probabilities. In *International Conference on Robotics and Automation (ICRA)*. IEEE, 1710–1715.

[8] K. O. Arras, B. Lau, S. Grzonka, M. Luber, O. M. Mozos, D. Meyer-Delius, and W. Burgard. 2012. Range-Based People Detection and Tracking for Socially Enabled Service Robots. *Towards Service Robots for Everyday Environments* 76 (2012), 235–280.

[9] T. M. Bagautdinov, A. Alahi, F. Fleuret, P. Fua, and S. Savarese. 2017. Social Scene Understanding: End-to-End Multi-Person Action Localization and Collective Activity Recognition.. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 3425–3434.

[10] J. C. Baxter. 1970. Interpersonal spacing in natural settings. *Sociometry* (1970), 444–456.

[11] L. Bazzani, M. Cristani, and V. Murino. 2012. Decentralized particle filter for joint individual-group tracking. In *Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 1886–1893.

[12] L. Bazzani, M. Zanotto, M. Cristani, and V. Murino. 2015. Joint individual-group modeling for tracking. *IEEE Transactions on Pattern Analysis & Machine Intelligence* 4 (2015), 746–759.

[13] G. Bertasius, H. S. Park, Yu. S., and J. Shi. 2017. First-Person Action-Object Detection with EgoNet. In *Proceedings of Robotics: Science and Systems (RSS)*. Cambridge, Massachusetts. https://doi.org/10.15607/RSS.2017.XIII.012

[14] A. Betancourt, P. Morerio, C. S. Regazzoni, and M. Rauterberg. 2015. The evolution of first person vision methods: A survey. *Transactions on Circuits and Systems for Video Technology* 25, 5 (2015), 744–760.

[15] V. Bettadapura, I. Essa, and C. Pantofaru. 2015. Egocentric field-of-view localization using first-person point-of-view devices. In *Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 626–633.

[16] D. Brščić, F. Zanlungo, and T. Kanda. 2017. Modelling of pedestrian groups and application to group recognition. In *40th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*. IEEE, 564–569.

[17] K. Caine, S. Šabanović, and M. Carter. 2010. Older Adults Engage in Privacy Enhancing Behaviors in a Home Monitored With Robots or Cameras. (2010).

[18] J. Carreira and C. Sminchisescu. 2011. Cpmc: Automatic object segmentation using constrained parametric min-cuts. *IEEE Transactions on Pattern Analysis & Machine Intelligence* 7 (2011), 1312–1328.

[19] I. Chamveha, Y. Sugano, Y. Sato, and A. Sugimoto. 2013. Social Group Discovery from Surveillance Videos: A Data-Driven Approach with Attention-Based Cues.. In *British Machine Vision Conference (BMVC)*.

[20] D. Chan, A. Taylor, and L. D. Riek. 2017. Faster Robot Perception Using Salient Depth Partitioning. In *Proceedings of the International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 4152–4158.

[21] Ming-Ching Chang, Nils Krahnstoever, and Weina Ge. 2011. Probabilistic group-level motion analysis and scenario recognition. In *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 747–754.

[22] W. Chang, J. P. White, J. Park, A. Holm, and S. Šabanović. 2012. The effect of group size on people's attitudes and cooperative behaviors toward robots in interactive gameplay. In *International Symposium on Robot and Human Interactive Communication (RO-MAN)*. IEEE, 845–850.

[23] I. Chatterjee and A. Steinfeld. 2015. Low Cost Perception of Dense Moving Crowd Clusters for Appropriate Navigation. In *, IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Workshop on Social Norms in Robotics and HRI.*

[24] W. Choi, Y. W. Chao, C. Pantofaru, and S. Savarese. 2014. Discovering groups of people in images. In *European conference on computer vision*. Springer, 417–433.

[25] W. Choi and S. Savarese. 2012. A unified framework for multi-target tracking and collective activity recognition. In *European Conference on Computer Vision*. Springer, 215–230.

[26] W. Choi, K. Shahid, and S. Savarese. 2009. What are they doing?: Collective activity classification using spatio-temporal relationship among people. In *12th International Conference on Computer Vision Workshops, ICCV Workshops*. IEEE, 1282–1289.

[27] W. Choi, K. Shahid, and S. Savarese. 2011. Learning context for collective activity recognition. In *CVPR 2011*. IEEE, 3273–3280.

[28] H. Christensen. 2016. A Roadmap for US Robotics from Internet to Robotics, 2016 Edition. *Sponsored by National Science Foundation & University of California, San Diego* (2016).

[29] F. Correia, S. Petisca, P. Alves-Oliveira, T. Ribeiro, F. Melo, and A. Paiva. 2017. Groups of humans and robots: Understanding membership preferences and team formation. In *Proceedings of Robotics: Science and Systems (RSS)*. Cambridge, Massachusetts. https://doi.org/10.15607/RSS.2017.XIII.024

[30] F. Correia, S. Petisca, P. Alves-Oliveira, T. Ribeiro, F. S. Melo, and A. Paiva. 2017. Groups of humans and robots: understanding membership preferences and team formation. *Proceedings of Robotics: Science and Systems (RSS)* (2017).

[31] N. Dalal and B. Triggs. 2005. Histograms of oriented gradients for human detection. In *Computer Society Conference on Computer Vision and Pattern Recognition.*, Vol. 1. IEEE, 886–893.

[32] J. Deng, A. Berg, S. Satheesh, H. Su, A. Khosla, and L. Fei-Fei. 2012. Imagenet large scale visual recognition competition 2012 (ILSVRC2012). *See net. org/challenges/LSVRC* (2012).

[33] J. Deng, W. Dong, R. Socher, L. Li, K. Li, and L. Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition (CVPR)*. Ieee, 248–255.

[34] Z. Deng, A. Vahdat, H. Hu, and G. Mori. 2016. Structure inference machines: Recurrent neural networks for analyzing relations in group activity recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 4772–4781.

[35] Z. Deng, M. Zhai, L. Chen, Y. Liu, S. Muralidharan, M. J. Roshtkhari, and G Mori. 2015. Deep structured models for group activity recognition. *arXiv preprint arXiv:1506.04191* (2015).

[36] P. Dollar, C. Wojek, B. Schiele, and P. Perona. 2012. Pedestrian detection: An evaluation of the state of the art. *Transactions on Pattern Analysis and Machine Intelligence* 34, 4 (2012), 743–761.

[37] I. Endres and D. Hoiem. 2010. Category independent object proposals. In *European Conference on Computer Vision*. Springer, 575–588.

[38] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. 2010. Object detection with discriminatively trained part-based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32, 9 (2010), 1627–1645.

[39] L. Feng and B. Bhanu. 2015. Tracking people by evolving social groups: an approach with social network perspective. In *2015 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 109–116.

[40] M. Fiore, H. Khambhaita, G. Milliez, and R. Alami. 2015. An adaptive and proactive human-aware robot guide. In *International Conference on Social Robotics*. Springer, 194–203.

[41] F. Flohr, D. Gavrila, et al. 2013. PedCut: An iterative framework for pedestrian segmentation combining shape models and multiple data cues.. In *Proceedings of the British Machine Vision Conference (BMVC)*.

[42] P. Foggia, G. Percannella, A. Saggese, and M. Vento. 2013. Real-time tracking of single people and groups simultaneously by contextual graph-based reasoning dealing complex occlusions. In *International Workshop on Performance Evaluation of Tracking and Surveillance (PETS)*. IEEE, 29–36.

[43] D. Fortun, P. Bouthemy, and C. Kervrann. 2015. Optical flow modeling and computation: a survey. *Computer Vision and Image Understanding* 134 (2015), 1–21.

[44] M. Fraune, F. Eyssel, M. F. Jung, and S. Šabanović. 2017. 3rd Workshop on Groups in Human-Robot Interaction. In *26th International Symposium on Robot and Human Interactive Communication*. IEEE.

[45] C. Garate, S. Zaidenberg, J. Badie, and F. Bremond. 2014. Group tracking and behavior recognition in long video surveillance sequences. In *International Conference on Computer Vision Theory and Applications (VISAPP)*, Vol. 2. IEEE, 396–402.

[46] A. Garrell and A. Sanfeliu. 2012. Cooperative social robots to accompany groups of people. *The International Journal of Robotics Research* 31, 13 (2012), 1675–1701.

[47] W. Ge, R. T. Collins, and R. B. Ruback. 2012. Vision-based analysis of small groups in pedestrian crowds. *IEEE transactions on pattern analysis and machine intelligence* 34, 5 (2012), 1003–1016.

[48] G. Gennari and G. D. Hager. 2004. Probabilistic data association methods in visual tracking of groups. In *Proceedings of the Computer Vision and Pattern Recognition (CVPR)*, Vol. 2. IEEE, II–II.

[49] R. Girshick. 2015. Fast r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision*. 1440–1448.

[50] R. Girshick, J. Donahue, T. Darrell, and J. Malik. 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*. 580–587.

[51] M. A. Goodrich and A. C. Schultz. 2007. Human-robot interaction: a survey. *Foundations and Trends in Human-Computer Interaction* 1, 3 (2007), 203–275.

[52] E. T. Hall. 1966. The hidden dimension. (1966).

[53] P. A. Hancock, D. R. Billings, K. E. Schaefer, Jessie Y.C. Chen, E. J. De Visser, and R. Parasuraman. 2011. A meta-analysis of factors affecting trust in human-robot interaction. *Human factors* 53, 5 (2011), 517–527.

[54] M. Hanheide, D. Hebesberger, T. Krajnik, et al. 2017. The when, where, and how: an adaptive robotic info-terminal for care home residents–a long-term study. In *Proceedings of the 2017 ACM/IEEE International Conference on Human-Robot Interaction*. ACM/IEEE.

[55] C. J. Hayes, M. Moosaei, and L. D. Riek. 2016. Exploring implicit human responses to robot mistakes in a learning from demonstration task. In *25th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*. IEEE, 246–252.

[56] K. He, X. Zhang, S. Ren, and J. Sun. 2014. Spatial pyramid pooling in deep convolutional networks for visual recognition. In *European conference on computer vision*. Springer, 346–361.

[57] M. S. Ibrahim, S. Muralidharan, Z. Deng, A. Vahdat, and G. Mori. 2016. A hierarchical deep temporal model for group activity recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1971–1980.

[58] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox. 2017. FlowNet 2.0: Evolution of Optical Flow Estimation with Deep Networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[59] Tariq Iqbal, Samantha Rack, and Laurel D Riek. 2016. Movement coordination in human–robot teams: a dynamical systems approach. *IEEE Transactions on Robotics (T-RO)* 32, 4 (2016), 909–919.

[60] T. Iqbal and L. D. Riek. 2016. A method for automatic detection of psychomotor entrainment. *IEEE Transactions on Affective Computing* 7, 1 (2016), 3–16.

[61] T. Iqbal and L. D. Riek. 2017. Coordination Dynamics in Multihuman Multirobot Teams. *IEEE Robotics and Automation Letters* 2, 3 (2017), 1712–1717.

[62] Tariq Iqbal and Laurel D Riek. 2017. Human-robot teaming: Approaches from joint action and dynamical systems. *Humanoid Robotics: A Reference* (2017), 2293–2312.

[63] O. H. Jafari, D. Mitzel, and B. Leibe. 2014. Real-time RGB-D based people detection and tracking for mobile robots and head-worn cameras. In *International Conference on Robotics and Automation (ICRA)*. IEEE, 5636–5643.

[64] R. Jonker and A. Volgenant. 1987. A shortest augmenting path algorithm for dense and sparse linear assignment problems. *Computing* 38, 4 (1987), 325–340.

[65] M. F. Jung, N. Martelaro, and P. J. Hinds. 2015. Using robots to moderate team conflict: the case of repairing violations. In *Proceedings of the Tenth Annual ACM/IEEE International Conference on Human-Robot Interaction*. ACM, 229–236.

[66] M. F. Jung, S. Šabanović, F. Eyssel, and M. Fraune. 2017. Robots in Groups and Teams. In *Companion of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing (CSCW '17 Companion)*. ACM, New York, NY, USA, 401–407. https://doi.org/10.1145/3022198.3022659

[67] M. E. Kaminski, M. Rueben, W. D. Smart, and C. M. Grimm. 2017. Averting Robot Eyes. *Maryland Law Review* 76, 4 (2017), 983.

[68] D. Karreman, G. Ludden, B. van Dijk, and V. Evers. 2015. How Can a Tour Guide Robot's Orientation Influence Visitors' Orientation and Formations? (2015), 7–14.

[69] A. Kendon. 1990. *Conducting Interaction: Patterns of behavior in focused encounters*. Vol. 7. CUP Archive.

[70] S. D. Khan, G. Vizzari, S. Bandini, and S. Basalamah. 2015. Detection of Social Groups in Pedestrian Crowds Using Computer Vision. In *Advanced Concepts for Intelligent Vision Systems*. Springer.

[71] A. Krizhevsky, I. Sutskever, and G. E. Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*. 1097–1105.

[72] A. Kubota, T. Iqbal, J. A. Shah, and L. D. Riek. 2019. Activity recognition in manufacturing: The roles of motion capture and sEMG+ inertial wearables in detecting fine vs. gross motion. *International Conference of Robotics and Automation (ICRA)* (2019).

[73] H. Kuzuoka, Y. Suzuki, J. Yamashita, and K. Yamazaki. 2010. Reconfiguring spatial formation arrangement by robot body orientation. In *Proceedings of the 5th ACM/IEEE International Conference on Human-Robot Interaction*. IEEE Press, 285–292.

[74] Kevin Lai, Liefeng Bo, and Dieter Fox. 2014. Unsupervised feature learning for 3d scene labeling. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 3050–3057.

[75] B. Lau, K. O. Arras, and W. Burgard. 2009. Tracking groups of people with a multi-model hypothesis tracker. In *International Conference on Robotics and Automation (ICRA)*. IEEE, 3180–3185.

[76] Bo. Lau, K. O. Arras, and W. Burgard. 2010. Multi-model hypothesis group tracking and group size estimation. *International Journal of Social Robotics* 2, 1 (2010), 19–30.

[77] A. LaViers, L. Bai, M. Bashiri, G. Heddy, and Y. Sheng. 2016. Abstractions for design-by-humans of heterogeneous behaviors. In *Dance Notations and Robot Motion*. Springer, 237–262.

[78] L. Leal-Taixé, G. Pons-Moll, and B. Rosenhahn. 2011. Everybody needs somebody: Modeling social and grouping behavior on a linear programming multiple people tracker. In *Computer Vision Workshops (ICCV)*. IEEE.

[79] H. R. Lee, S. Šabanović, W. Chang, S. Nagata, J. Piatt, C. Bennett, and D. Hakken. 2017. Steps Toward Participatory Design of Social Robots: Mutual Learning with Older Adults with Depression. In *Proceedings of the 2017 ACM/IEEE International Conference on Human-Robot Interaction*. ACM, 244–253.
[80] A. Lerner, Y. Chrysanthou, and D. Lischinski. 2007. Crowds by example. In *Computer graphics forum*, Vol. 26. Wiley Online Library, 655–664.
[81] N. Li, Y. Zhang, W. Luo, and N. Guo. 2017. Instant coherent group motion filtering by group motion representations. *Neurocomputing* 266 (2017), 304–314.
[82] T. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. 2014. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*. Springer, 740–755.
[83] T. Linder and K. O. Arras. 2014. Multi-model hypothesis tracking of groups of people in RGB-D data. In *17th International Conference on Information Fusion (FUSION)*. IEEE, 1–7.
[84] T. Linder, S. Breuers, B. Leibe, and K. O. Arras. 2016. On multi-modal people tracking from mobile platforms in very crowded and dynamic environments. In *International Conference on Robotics and Automation (ICRA)*. IEEE, 5512–5519.
[85] T. Lorenz, A. Mortl, B. Vlaskamp, A. Schubo, and S. Hirche. 2011. Synchronization in a Goal-Directed Task: Human Movement Coordination with Each Other and Robotic Partners. *International Symposium on Robot and Human Interactive Communication (RO-MAN)* (2011).
[86] M. Luber and K. O. Arras. 2013. Multi-hypothesis social grouping and tracking for mobile robots.. In *Robotics: Science and systems (RSS)*.
[87] S. Matsumoto and L. D. Riek. 2019. Fluent Coordination in Proximate Human Robot Teaming. *In Proceedings of the Robotics, Science, and Systems (RSS) Workshop on AI and Its Alternatives for Shared Autonomy in Assistive and Collaborative Robotics* (2019).
[88] R. Mazzon, F. Poiesi, and A. Cavallaro. 2013. Detection and tracking of groups in crowd. In *International Conference on Advanced Video and Signal Based Surveillance (AVSS)*. IEEE, 202–207.
[89] R. Mead and M. J. Matarić. 2017. Autonomous human–robot proxemics: socially aware navigation based on interaction potential. *Autonomous Robots* 41, 5 (2017), 1189–1201.
[90] M. Moussaïd, N. Perozo, S. Garnier, D. Helbing, and G. Theraulaz. 2010. The walking behaviour of pedestrian social groups and its impact on crowd dynamics. *PloS one* 5, 4 (2010), e10047.
[91] M. Mucientes and W. Burgard. 2006. Multiple hypothesis tracking of clusters of people. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 692–697.
[92] J. Mumm and B. Mutlu. 2011. Human-robot proxemics: physical and psychological distancing in human-robot interaction. In *Proceedings of the 6th International Conference on Human-Robot Interaction*. ACM, 331–338.
[93] M. Munaro, F. Basso, and E. Menegatti. 2012. Tracking people within groups with RGB-D data. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2101–2107.
[94] J. Nigam and R. M. Rameshan. 2017. EgoTracker: Pedestrian Tracking with Re-identification in Egocentric Videos. In *Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. IEEE, 980–987.
[95] N. Noceti and F. Odone. 2014. Humans in groups: The importance of contextual information for understanding collective activities. *Pattern Recognition* 47, 11 (2014), 3535–3551.
[96] M. O'Connor and L.D. Riek. 2015. Detecting Social Context: A Method for Social Event Classification Using Naturalistic Multimodal Data. *In Proceedings of the 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)* (2015).
[97] M. Okamoto and K. Yanai. 2013. Summarization of egocentric moving videos for generating walking route guidance. In *Pacific-Rim Symposium on Image and Video Technology*. Springer, 431–442.
[98] S. Pellegrini, A. Ess, K. Schindler, and L. Van Gool. 2009. You'll never walk alone: Modeling social behavior for multi-target tracking. In *Computer Vision, 2009 IEEE 12th International Conference on*. IEEE, 261–268.
[99] S. Pellegrini, A. Ess, and L. Van Gool. 2010. Improving data association by joint modeling of pedestrian trajectories and groupings. In *Computer Vision*. Springer.
[100] M. Piccardi. 2004. Background subtraction techniques: a review. In *International Conference on Systems, Man and Cybernetics*, Vol. 4. IEEE, 3099–3104.
[101] Z. Qin and C. R. Shelton. 2012. Improving multi-target tracking via social grouping. In *Computer Vision and Pattern Recognition (CVPR)*. IEEE.
[102] O. A. I. Ramírez, G. Varni, M. Andries, M. Chetouani, and R. Chatila. 2016. Modeling the dynamics of individual behaviors for group detection in crowds using low-level features. In *5th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*. IEEE, 1104–1111.
[103] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. 2016. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 779–788.
[104] J. Redmon and A. Farhadi. 2017. YOLO9000: better, faster, stronger. *arXiv preprint* (2017).
[105] J. Redmon and A. Farhadi. 2018. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767* (2018).

[106] S. Ren, K. He, R. Girshick, and J. Sun. 2015. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems*. 91–99.
[107] L.D. Riek. 2013. The Social Co-Robotics Problem Space: Six Key Challenges. *In Proceedings of Robotics: Science, and Systems (RSS), Robotics Challenges and Visions* (2013).
[108] L. D. Riek. 2015. Robotics technology in mental health care. *Artificial Intelligence in Behavioral and Mental Health Care* (2015), 185.
[109] L. D. Riek. 2017. Healthcare robotics. *Commun. ACM* (2017), 68–78.
[110] S. Rossi and P. D'Alterio. 2017. Gaze Behavioral Adaptation Towards Group Members for Providing Effective Recommendations. In *International Conference on Social Robotics*. Springer, 231–241.
[111] M. S. Ryoo and J.K. Aggarwal. 2010. UT-interaction dataset, ICPR contest on semantic description of human activities (SDHA). In *IEEE International Conference on Pattern Recognition Workshops*, Vol. 2. 4.
[112] M. S. Ryoo, T. J. Fuchs, L. Xia, J. K. Aggarwal, and L. Matthies. 2015. Robot-centric activity prediction from first-person videos: What will they do to me?. In *Proceedings of the Tenth Annual ACM/IEEE International Conference on Human-Robot Interaction*. ACM, 295–302.
[113] M. S. Ryoo and L. Matthies. 2013. First-person activity recognition: What are they doing to me?. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2730–2737.
[114] S. Šabanović, S. Reeder, and B. Kechavarzi. 2014. Designing robots in the wild: In situ prototype evaluation for a break management robot. *Journal of Human-Robot Interaction* 3, 1 (2014), 70–88.
[115] S. Schneider and F. Kümmert. 2016. Exercising with a humanoid companion is more effective than exercising alone. In *16th International Conference on Humanoid Robots (Humanoids)*. IEEE, 495–501.
[116] R. J. Sethi. 2015. Towards defining groups and crowds in video using the atomic group actions dataset. In *International Conference on Image Processing (ICIP)*. IEEE, 2925–2929.
[117] M. Shiomi, T. Kanda, S. Koizumi, H. Ishiguro, and N. Hagita. 2007. Group attention control for communication robots with wizard of OZ approach. In *Proceedings of the ACM/IEEE international Conference on Human-Robot Interaction*. ACM, 121–128.
[118] T. Shu, S. Todorovic, and S. Zhu. 2017. CERN: confidence-energy recurrent network for group activity recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Vol. 2. 5523–5531.
[119] J. Šochman and D. C. Hogg. 2011. Who knows who-inverting the social force model for finding groups. In *International Conference on Computer Vision Workshops (ICCV)*. IEEE, 830–837.
[120] F. Solera and S. Calderara. 2013. Social groups detection in crowd through shape-augmented structured learning. In *International Conference on Image Analysis and Processing*. Springer, 542–551.
[121] F. Solera, S. Calderara, and R. Cucchiara. 2016. Socially constrained structural learning for groups detection in crowd. *IEEE transactions on pattern analysis and machine intelligence* 38, 5 (2016), 995–1008.
[122] L. Spinello and K. O. Arras. 2011. People detection in RGB-D data. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 3838–3843.
[123] P. Stefano, E. Andreas, S. Konrad, and L. Van Gool. 2009. You'll never walk alone: Modeling social behavior for multi-target tracking. In *International Conference on Computer Vision Workshops (ICCV)*.
[124] P. Sudowe and B. Leibe. 2011. Efficient Use of Geometric Constraints for Sliding-Window Object Detection in Video.. In *International Conference on Computer Vision Systems (ICVS)*. Springer, 11–20.
[125] J. Sung, H. I. Christensen, and R. E. Grinter. 2009. Robots in the wild: understanding long-term use. In *Proceedings of the 4th ACM/IEEE International Conference on Human Robot Interaction*. ACM, 45–52.
[126] L. Takayama and C. Pantofaru. 2009. Influences on proxemic behaviors in human-robot interaction. In *International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 5495–5502.
[127] A. Taylor, H. Lee, A. Kubota, and L. D. Riek. 2019. Coordinating Clinical Teams: Using Robots to EmpowerNurses to Stop the Line. *In Proceedings of Computer Supported Cooperative Work (CSCW)* (2019).
[128] A. Taylor, H. R. Lee, A. Kubota, and L. D. Riek. 2019. Coordinating clinical teams: Using robots to empower nurses to stop the line. *Proceedings of the ACM on Human-Computer Interaction* 3, CSCW (2019), 221.
[129] A. Taylor and L.D. Riek. 2019. Group Perception Methods to Support Human-Robot Teaming. In *Southern California Robotics Symposium (SCR)*.
[130] A. Taylor and L. D. Riek. 2016. Robot Perception of Human Groups in the Real World: State of the Art. In *AAAI Fall Symposium Series: Artificial Intelligence for Human-Robot Interaction Technical Report*, Vol. 4. 2017.
[131] A. Taylor and L. D. Riek. 2018. Robot-Centric Human Group Detection. In *13th Annual ACM/IEEE International Conference on Human-Robot Interaction, Social Robots in the Wild Workshop*. IEEE.
[132] R. Triebel, K. Arras, R. Alami, L. Beyer, S. Breuers, R. Chatila, M. Chetouani, D. Cremers, V. Evers, M. Fiore, et al. 2016. Spencer: A socially aware service robot for passenger guidance and help in busy airports. In *Field and Service Robotics*. Springer, 607–622.
[133] S. Tseng, Y. Chao, C. Lin, and L. Fu. 2016. Service robots: System design for tracking people through data fusion and initiating interaction with the human group by inferring social situations. *Robotics and Autonomous Systems (RSS)* 83 (2016), 188–202.

[134] M. Vázquez, E. J. Carter, B. McDorman, J. Forlizzi, A. Steinfeld, and S. E. Hudson. 2017. Towards Robot Autonomy in Group Conversations: Understanding the Effects of Body Orientation and Gaze. In *Proceedings of the 2017 ACM/IEEE International Conference on Human-Robot Interaction (HRI '17)*. ACM, New York, NY, USA, 42–52. https://doi.org/10.1145/2909824.3020207

[135] M. Vázquez, E. J. Carter, J. A. Vaz, J. Forlizzi, A. Steinfeld, and S. E. Hudson. 2015. Social group interactions in a role-playing game. In *Proceedings of the Tenth Annual International Conference on Human-Robot Interaction Extended Abstracts*. ACM, 9–10.

[136] M. Vázquez, A. Steinfeld, and S. E. Hudson. 2015. Parallel detection of conversational groups of free-standing people and tracking of their lower-body orientation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 3010–3017.

[137] J. Vroon, M. Joosse, M. Lohse, J. Kolkmeier, J. Kim, K. Truong, G. Englebienne, D. Heylen, and V. Evers. 2015. Dynamics of social positioning patterns in group-robot interactions. In *24th International Symposium on Robot and Human Interactive Communication (RO-MAN)*. IEEE, 394–399.

[138] X. Wang, M. Wang, and W. Li. 2014. Scene-specific pedestrian detection for static video surveillance. *Transactions on Pattern Analysis and Machine Intelligence* 36, 2 (2014), 361–374.

[139] N. Wojke, A. Bewley, and D. Paulus. 2017. Simple online and realtime tracking with a deep association metric. In *2017 IEEE International Conference on Image Processing (ICIP)*. IEEE, 3645–3649.

[140] S. A. Yang, E. Gamborino, C. T. Yang, and L. C. Fu. 2017. A study on the social acceptance of a robot in a multi-human interaction using an F-formation based motion model. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2766–2771. https://doi.org/10.1109/IROS.2017.8206105

[141] H. Yu, Y. Zhou, J. Simmons, C. P. Przybyla, Y. Lin, X. Fan, Y. Mi, and S. Wang. 2016. Groupwise tracking of crowded similar-appearance targets from low-continuity image sequences. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 952–960.

[142] F. Yücel, Z.and Zanlungo, T. Ikeda, T. Miyashita, and N. Hagita. 2013. Deciphering the crowd: Modeling and identification of pedestrian group motion. *Sensors* 13, 1 (2013), 875–897.

[143] S. Zaidenberg, B. Boulay, C. Garate, D. P. Chau, E. Corvée, and F. Bremond. 2011. Group interaction and group tracking for video-surveillance in underground railway stations. In *International Workshop on Behaviour Analysis and Video Understanding (ICVS 2011)*. 10.

[144] M. Zanotto, L. Bazzani, M. Cristani, and V. Murino. 2012. Online bayesian nonparametrics for group detection. In *Proceedings of British Machine Vision Conference*. 111–1.

[145] S. Zhang, R. Benenson, M. Omran, J. Hosang, and B. Schiele. 2016. How far are we from solving pedestrian detection?. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1259–1267.

[146] Y. Zhang, L. Qin, S. Zhang, H. Yao, and Q. Huang. 2015. Formation period matters: Towards socially consistent group detection via dense subgraph seeking. In *Proceedings of the 5th ACM on International Conference on Multimedia Retrieval*. ACM, 475–478.

[147] F. Zhong and J. Zhang. 2013. Linear discriminant analysis based on L1-norm maximization. *IEEE Transactions on Image Processing* 22, 8 (2013), 3018–3027.