

MASTER

Design and Evaluation of Strategies for Multi-Human Intention Prediction in Multi-Robot Systems

van der Vorst, Sander W.F.

Award date:
2023

[Link to publication](#)

Disclaimer

This document contains a student thesis (bachelor's or master's), as authored by a student at Eindhoven University of Technology. Student theses are made available in the TU/e repository upon obtaining the required degree. The grade received is not published on the document as presented in the repository. The required complexity or quality of research of student theses may vary by program, and the required minimum study period may vary in duration.

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain

Design and Evaluation of Strategies for Multi-Human Intention Prediction in Multi-Robot Systems

GRADUATION PROJECT

MASTER ARTIFICIAL INTELLIGENCE & ENGINEERING SYSTEMS
CST 2022.022

Name	Student number
Vorst, S.W.F. van der	0996043

Supervisors

Torta, E.
Verhees, E.D.T.

Committee Members

Kuling, I.A.
Molengraft, M.J.G. van de

Abstract— This work presents a pipeline to infer semantic goals of multiple humans based on their motion cues as perceived by multiple robots. Experiments were conducted to gather data in order to evaluate the performance of the proposed pipeline. Lastly, the improvements on the human intention prediction by sharing data between robots are identified and compared to the individual cases of the robots. This thesis aims to take a first step to achieve multi-human multi-robot collaboration.

Index Terms—Strategy Evaluation, Intention Prediction, Multi-Human, Multi-Robot, Semantic Goals, Collaboration

I. INTRODUCTION

In the field of warehousing, automation is one of the key drivers for increasing efficiency. The integration of Automated Vehicles (AV's) and other types of robots into the core warehouse processes has rapidly accelerated in the past decade [1] [2] [3]. But the industry slowly gravitates away from the stationary automated warehouses towards a more flexible and scalable solution, namely a fleet of Automated Vehicles with increased decentralized decision-making capabilities [4] [5]. However, until every task can be automated, robots and humans will unavoidably be present in the same working space. Society, and especially the warehousing business, will move towards a situation where work is shared between multiple robots and multiple humans whom have to collaborate on their everyday tasks to achieve lower labour intensity and higher productivity and profitability. Current research, such as the ILIAD project [2] and Vanderlande's Pallet AV project [3], focus mainly on a centrally controlled fleet of AV's that have precalculated trajectories and goals throughout the warehouse. These AV's currently have limited human intention prediction capabilities, because this is mainly used for collision avoidance and enforcing general safety rules instead of human-robot collaboration. Due to the expected increasing decentralization of fleet control, mobile robots not only need to be able to estimate the semantic goals of multiple humans in their vicinity, but also make choices to work with these humans towards that same goal. This thesis aims to take a first step to achieve multi-human multi-robot collaboration by identifying how and why the sharing of data between robots can lead to better estimation of human intentions.

A noticeable trend observed at mayor players in the warehousing field is the transition of the current order picking solutions to a situation where humans and robots are working together, which can result in a more cost-effective and easily scalable situation. Often small business in Europe or in other less developed parts of the world do not have the money or space to incorporate a fully automated warehouse system. Such a system has a high initial installation cost and often needs to be build around the peak throughput, meaning that such a system operates beneath maximum capacity most of the year. It is thus in the interest of these small business to have a relatively compact system that has lower installation costs and higher scalability when the business grows or a peak in orders needs to be processed. Both can be achieved

by designing a fleet controller to calculate only the basic order of tasks and routes for each robot in the fleet and retain more autonomy for each individual robot. This increasingly decentralized structure has an additional advantages that the amount of active robots can easily be scaled up or down based on the amount of work and that the system is more robust to individual failing robots. These AV's can only transport products around the warehouse, like automated shopping carts riding from the loading dock through the warehouse and back. This circumvents a problem that is very hard to solve, namely the construction of a robot that can pick a variety of different products from shelves at different heights and distances. Humans are responsible for the picking of orders in a designated corridor of shelves, and putting them inside the container of an AV. In this situation, multiple humans are walking in the same area as where the robots operate. It would thus be very beneficial for the AV's to be able to estimate to which goal a human is walking, so it may for example adjust its own trajectory or current goal based on where the human is going. Furthermore, the AV's need to communicate and share their observations to cover a bigger part of the total warehouse and thus make more informed decisions. In order to achieve these functionalities, the AV's need to be able to translate a sensory data such as a video feed or infra-red depth information into 2D trajectories of each human in its field of view and relate the trajectories to the possible goals in their vicinity. Next, they need to evaluate which goals the humans are most likely to be moving towards and find a consensus on the situation together. A pipeline on how to infer human semantic goals based on the robot perception of their motion is proposed in this thesis.

A comparable situation, in the sense that intention prediction and collaboration in a multi-human multi-robot situation is becoming more relevant, is during the robot-soccer matches in the MSL RoboCup. At the world championship in 2022, for the first time a human was allowed to play with the robots during a match [6]. A soccer match is a highly collaborative environment and being able to estimate what the human on their own team is intending to achieve as well as what the human on the enemy team is intending to do is important for planning strategies and, in general, playing well together. In order for this research to also be relevant in a soccer robot case, the test setup used for gathering data is inspired by similar human-robot collaboration characteristics from both the warehouse and soccer robot use cases. Due to the similarity between these cases, a similar pipeline can be utilised in both cases.

This thesis is structured as follows. In Section II the rules and constraints of the experiments used to gather data are explained. These experiments are designed to incorporate some characteristics that are relevant for human-robot teams in warehousing situations such as retrieving a package and bringing it back to the robot. The experiments are performed with two humans, two robots and four goals. In Section III the proposed Goal Estimation Pipeline is discussed. First the

video and depth data of the camera are converted into human joint locations in a 3D environment. Then these joint locations are translated into a sequence of 2D positions and orientations over time. Lastly, a 2D-Gaussian function is used to estimate the goal of both humans for each robot individually. Their respective estimation is compared to the situation where the estimations of both robots are combined to investigate if and where this can lead to improvements. In Section IV it is explained how the experiments were conducted and in Section V the performance of each part of the proposed Goal Estimation Pipeline is discussed. Lastly in Section VI the conclusions are drawn up and in Section VII the recommendations for further research are formulated.

A. Related Work

Research in the field of intention prediction can be subdivided into three different areas: current behavior estimation, trajectory prediction and semantic goal prediction. The goal of the models used is to provide the agent with an accurate assessment of a specific situation so appropriate actions can be taken. The first category focuses on estimating the current actions the tracked entity is engaged in, for example determining if a human is showing behavior related to crossing a street by feeding raw data from the camera of an autonomous driving vehicle into a deep neural network [7], or by using an LSTM-network (a form of a recursive neural network) to categorize a persons behavior inside a working environment by detecting if human is engaging with another human or just walking by [8]. These (deep) neural network models can often be opaque in their inner workings, but can however provide quite accurate results. Next to the direct actions of humans, also the actions of objects like cars can be analyzed to determine if for example lane changing behavior is present based on positional data [9].

The next category of intention prediction is the extrapolation of trajectories of moving objects or humans. For example, the past path of a human in combination with the proximity of other humans in its vicinity can be analyzed and used as input for an LSTM-network. The future trajectory of a human in a dense crowd can be accurately estimated with the proposed method from [10]. These trajectory estimation algorithms are also used in research affiliated with self-driving vehicles to extrapolate the trajectories of pedestrians to avoid collisions [11].

The last category of intention prediction is the estimation of semantic goals. This can be done in abstract ways like analyzing clicking behavior of a human in an online webshop to predict what the customer wants to buy in order to show personally tailored discounts [12]. Less abstract methods involve using state estimation algorithms like a Hidden Markov model to analyze car movement and determine to which lane a car will change [13], or using human pose estimation to predict the intentions of workers in warehouse corridors towards predefined semantic goals [14]. This main focus of this thesis is the latter category, namely the intention

prediction of semantic goals from user movement.

Recent research on semantic goal estimation includes [15], where the freely available human pose estimation library OpenPose [16] is used to obtain the position and orientation of a human. Bayesian mathematics is then used to estimate the leading hypothesis corresponding to a semantic goal. In this research a single robot is used to estimate the goal of a single human. The goals consists of areas in a T-split corridor and areas around the robot itself. In this work, the methods described in [15] are extended to incorporate multiple humans and estimate the goals of each in an open area without walls. Furthermore, the influence of sharing data between both robots on the convergence to the correct goal is assessed.

The contribution of this thesis is thus a method to estimate the semantic goals of multiple humans in a situation where multiple robots are present. A Bayesian approach is used to calculate the likelihood of each hypothesis based on the positions and orientations of each human. The influence of sharing data between robots is compared to the situation where both robots operate individually. Lastly, a repository¹ is provided containing a program with interchangeable parts that facilitates easily swapping and testing of different methods for data association, probability calculation and data sharing. The data gathered from the performed experiments is also available for future research.

II. EXPERIMENT SETUP

In order to test the proposed method, a series of experiments was designed to gather data that can be used to gain insight into the performance of the proposed pipeline and identify situations where the sharing of data leads to improved results. The experiments are designed such that they incorporate characteristics that are relevant for human-robot teams in a warehouse such as an open movement area with no walls or other physical obstructions, humans starting outside the field of view of one or both robots, overlapping fields of view of the robots and humans picking up parcels and bringing them back to the robot. In both a warehouse situation and the robot soccer match situation, robots and humans fulfill different rolls in the same team based on both their limitations. A robot is for example very well suited for moving parcels around following an optimal route, but it is very hard for a robot to pick up non-uniform parcels from different places. A human can easily select and pick-up a variety of different parcels with different weights and shapes, but cannot transport those effectively over long distances. By assigning the tasks best suited for both robots and humans, the highest efficiency can be reached. In the experiments the humans perform the task of walking towards a goal, picking it up and bringing it back to the robot. The task of the robot is only to observe and save data.

¹ https://gitlab.tue.nl/et_projects/svintentionrecognition

The experiments are conducted on one half of a robot soccer field, which is shown in Figure 1. The robots are recreated by two Kinect V2 camera's each connected to a laptop and are thus stationary. These camera's are placed a 2 meters apart, 1.5 meters outside the borders of the field. By choosing this setup, the middle of the field is covered by both robots while still allowing some areas to only be covered by one of the two robot. This is done to cover a bigger total part of the field and also create interesting situations to analyse the influence of sharing data. The field of view of both robots are visible in Figure 1.

The semantic goals are represented by four soccer balls that are placed in four different configurations in the field. The location of these goals is assumed to be known by the robots. For each configuration of starting positions for the balls, different starting positions for each human are devised. The exact combinations of starting positions for the goals and the humans can be found in Appendix B. They are however chosen in such a way that the humans start close to different goals and at different spots in the field, eliminating external influences as much as possible and enabling the emergence of interesting situations such as crossings and walking past another goal towards the next. In each experiment, two people perform 10 different trials. An experiment lasts about 20 minutes. All experiments are grouped such that the participants do not start at the same spot or with the same starting positions of the soccer balls in subsequent trials, again eliminating as much external influences as possible. The experiments are held during three subsequent days. People are allowed to participate more than once, but not more than once a day.

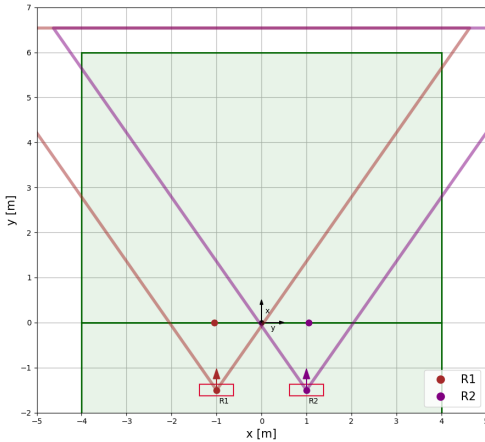


Figure 1: The empty field where the experiments are held. Both robots are displayed with their field of views and the location for putting down the picked-up ball.

The experiments are performed as follows: The goals and humans start at the pre-planned locations. The humans then receive the instructions to each walk towards a ball, pick it up and bring it towards one of the robots. The only constraints are that they cannot pick up the same ball and that they are not allowed to arrive at the same robot at the same time. After

they have put their ball at the designated spot before the robot, they have to exit the field to the side to not walk in front of the camera's. If the first human has exited the field, it is again allowed to potentially bring a ball to the same robot. If both persons have exited the field, the trial ends. The soccer balls are put at the next locations while the humans walk towards their next starting positions. This is repeated until all trials are done. It is aimed to perform about 100 experiments in this way in order to capture all possible interesting situations and also create some duplicates for redundancy. A contingency table B.1 of the combinations of starting positions of the soccer balls and starting positions of the humans can be found in Appendix B. An overview of the configurations used in each experiment can be found in Table D.13 in Appendix D. The data obtained with these experiments is then used as input for the Goal Estimation Pipeline explained in the next section.

III. GOAL ESTIMATION PIPELINE

In this section the complete method used to transform a frame from a Kinect camera into likelihood evaluations for all goals is explained. An overview of the full pipeline is shown in Figure 2. The method is divided into three parts: *Perception*, *Trajectory* and *Intention Prediction*. In Subsection III-A: *Perception*, it is explained how the color frame and pointcloud received from the Kinect are used to find the $[x, y, z]$ -coordinates of the six assessed joint positions by OpenPose. After that, in Subsection III-B: *Trajectory* it is shown how these 3D joint positions are converted into an $[x, y]$ -position and how the orientation θ is determined. Next, in Subsection III-C: *Intention Prediction* it is explained how the 2D position and orientation sequences are used to calculate the most likely goal of each human using a Bayesian approach. It is also described how the data of the two robots is shared. Lastly, an overview of the complete method is given with the help of a pseudo-code algorithm.

A. Perception

For the *Perception* part of the pipeline frames obtained from the Kinect camera are transformed into 3D joint positions in the field. The output of the Kinect consists of a RGB color image with a size of 424×512 and a corresponding pointcloud with x, y and z values for all pixels. The original resolution of the Kinect is HD 1080×1920 , but this is scaled down by the drivers of the Kinect to match the coarseness of the depth sensors. The conversion from depth values to a pointcloud is done by the drivers of the Kinect. The image frame is analyzed by the pose estimation algorithm OpenPose [17]. A brief summary of the workings of this free-to-use algorithm are explained in Subsection III-A1. The output of OpenPose consists of 2D pixel values for the tracked points, which are converted to world coordinates with the pointcloud from the Kinect. This process is explained in subsection III-A2.

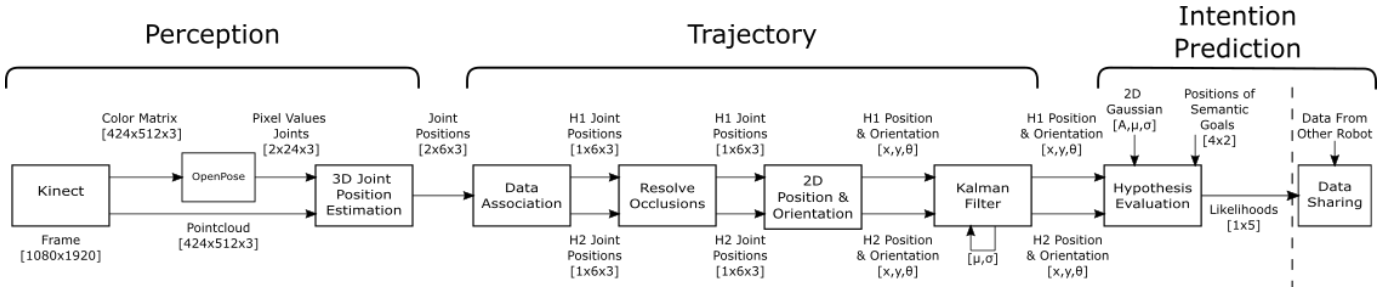


Figure 2: Visual representation of the Goal Estimation Pipeline for the processing of one frame. The part executed for each robot is subdivided into *Perception*, *Trajectory* and *Intention Prediction*. After that, the likelihoods of the hypotheses are shared between robots. The type of data communicated between parts is displayed above each arrow connecting the parts. Each part is discussed in Section III

1) *OpenPose*: OpenPose is an open-source library that can be used for real-time multi-person keypoint detection. The input of the model is a single frame of any size. The model consists of a two-branch multi-stage convolutional neural network, where one branch predicts confidence maps for particular body parts and the other branch estimates part-affinity fields. These fields indicate towards what directions different body parts are most likely oriented with respect to other body parts of the same human. Together these two branches are merged in a bipartite graph that matches associated body parts of multiple humans (if present in the frame). For an in depth explanation of the workings of the OpenPose model the reader is referred to [17]. The output is 24 2D pixel coordinates per human corresponding to major joints of the human body as well as some other points of interest indicating the orientation of the head. Next to that, also the confidence between 0 and 1 of each joint position is given. The confidence is only used in the occlusions part of the pipeline and thus not shown in the overview in Figure 2. An example of the result of OpenPose is displayed in Figure 3.

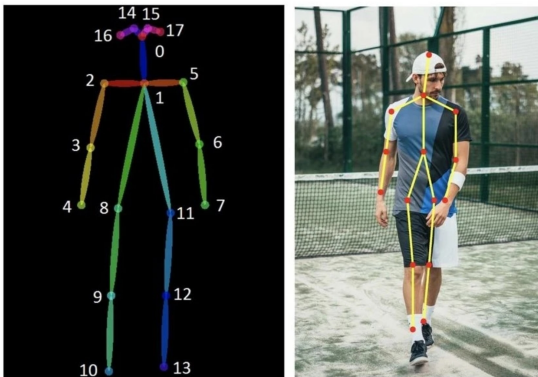
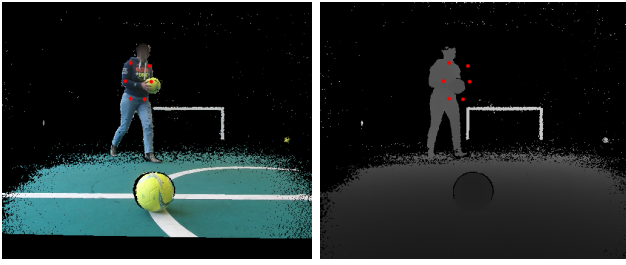


Figure 3: Keypoints detected by OpenPose on the COCO dataset. [16]

Only 6 of the 24 joint locations are used: the shoulders, the elbows and the hips. These joint pairs are indicated in Figure 3 by the numbers 2 & 5, 3 & 6 and 8 & 11

respectively. These three joint pairs are chosen because they remain most constant when a person is walking, while joints like the hands, knees and feet oscillate around the absolute center of the human (when seen from a side view). The average 2D location of the chosen six joints is thus closest to the average location of a person as a whole. In the next subsection it is explained how the pixel coordinates of the six joints are converted back to the world coordinates in which the positions of the humans are measured.

2) *3D Joint Positions*: The data received from the OpenPose algorithm consists of six joint locations in pixel coordinates $[p_x, p_y]$. These coordinates need to be converted to $[x, y, z]$ -coordinates with the help of the positions stored in the pointcloud matrix of the Kinect. The OpenPose algorithm outputs float values, so it cannot immediately be linked to one entry in the pointcloud. Furthermore, it is observed that pixel coordinates outputted by OpenPose in some cases do not match with the visual location of the joints in the frame, as illustrated in Figure 4. After inspection of multiple cases, the compression of the resolution from 1080×1920 to 424×512 by the Kinect is appointed as the cause of this mismatch between the color matrix and the pointcloud values. Algorithm 1 is thus devised to iteratively increase the search range in the pointcloud matrix in order to find enough suitable points for an estimate of the joint location.



(a) 424×512 color frame as received from the Kinect with the joint positions of the six followed joints. (b) 424×512 depth frame as received from the Kinect before conversion to world coordinates. A lighter color means that the target is closer to the camera.

Figure 4: Example of a mismatch between pixel values for the joint positions from OpenPose and the depth image received from the Kinect.

Algorithm 1 Compute $[x, y, z]$ -position of joint

Require: keypoint: $[p_x, p_y]$ in pixel coordinates

Require: pointcloud: all $[x, y, z]$ points in world coordinates

- 1: Iteration $i = 2$
 - 2: Max iterations $k = 8$
 - 3: Threshold $h = 5$
 - 4: **while** amount of valid points $\leq h$ **do**
 - 5: Take $[x, y, z]$ -values of $2i \times 2i$ square around keypoint
 - 6: Discard points with pixel values outside 424×512 frame
 - 7: Discard points with no depth value
 - 8: Calculate center point of depth values in world coordinates
 - 9: Discard points outside 90% surface under fitted normal distribution curve
 - 10: Discard points with norm of more than 1.0m to center point
 - 11: $i = i + 1$
 - 12: **if** $i \geq k$ **then**
 - 13: **return** Empty Array
 - 14: **end if**
 - 15: **end while**
 - 16: Recalculate center point
 - 17: **return** center point
-

The input of the algorithm is the keypoint $[p_x, p_y]$ from OpenPose and the complete pointcloud from the Kinect. In row 1-3 the parameters are initialized. The initial value of the iteration parameter i is chosen to be equal to 2. This means at least 16 pointcloud points are evaluated, which ensures enough points are available to identify possible outliers. The

parameters k and h determine the values of the evaluation in row 12-14 where the algorithm terminates if less than 5 suitable points are found after 8 iterations. In every iteration i , all points corresponding to the pixels in a square with width $2i$ around $[p_x, p_y]$ are evaluated. If one of the evaluated pixels lays outside the matrix, the corresponding point is discarded. This can happen when the location from OpenPose lays close to the border of the frame. All points outside the range of the Kinect depth sensor have no depth value and are thus also discarded. These actions take place in row 5-7. Next, the mean and standard deviation of the current point set is calculated. Points with an absolute distance to the mean of more than 1.7 times the standard deviation (corresponding to 90% of the surface under the fitted normal distribution curve) are discarded, as well as points located more than 1.0 meter from the mean. The depth values in the matrix can increase rigorously when compared to adjacent pixels, so it is essential to filter outliers effectively. These computations happen in row 8-11. In the last part of the algorithm the iteration parameter i is increased and it is evaluated if enough viable points were found. If that is the case, the mean point of the set is recalculated in row 16 and returned as output in row 17.

B. Trajectory

In this part of the pipeline the six corrected joint positions of each human from OpenPose are used to determine the average $[x, y]$ -position of each human. In Subsection III-B1 it is first explained how each group of six joint positions is added to the sequence of the corresponding human. OpenPose outputs data for both persons in the scene, but it cannot be known beforehand which data belongs to Human $H1$ and which data belongs to Human $H2$. It thus needs to be determined to which current sequence the new data most likely belongs. After that, in Subsection III-B2 it is described how missing joint positions are estimated. It is also explained how occlusions of joints by other body parts of the same human are detected and resolved with the newly proposed Far Joint Inferring algorithm (Alg. 4). Lastly, in Subsection III-B3 it is explained how the orientation θ of each human is calculated based on a weighted moving average of the past velocity vectors. Also a brief description is given of how a Kalman filter is used to adjust the found position and orientation based on the assumption of constant velocity.

1) *Data Association:* Each new set of joint positions needs to be associated to the correct set of already evaluated points. OpenPose does not always detect each human in the scene in the same order. This can result in the fact that the joint position groups are swapped with respect to the human sequences and thus it is needed to evaluate to which of the existing sequences the new joint positions most likely belong. This is shown in Algorithm 2. First, in row 4 a prediction of the joint position is made by calculating a weighted moving average of the five past joint positions. The moving average calculation is shown in Eq. (1), where x_{t+1} , y_{t+1} and z_{t+1} represent the prediction of the new joint position, N is the total amount of frames

that is looked back, n is the frame number and t is the current frame number. Taking a moving average ensures that each data point closer to the current frame weight twice as heavy as data points further back in the sequence.

$$\begin{bmatrix} x_{t+1} \\ y_{t+1} \\ z_{t+1} \end{bmatrix} = \frac{1}{\sum_{n=0}^{N-1} 2^{N-n}} \sum_{n=0}^{N-1} 2^{N-n} \cdot \begin{bmatrix} x_{t-n} \\ y_{t-n} \\ z_{t-n} \end{bmatrix}. \quad (1)$$

If it is the first frame, the initial starting position of the human is taken as the prediction value (row 5 – 6). Furthermore, if the moving average does not produce a viable result, for example because a human cannot be seen resulting in no data at that time frame, the last known position of that joint is taken (row 7 – 8). Next, in row 8 the absolute norm between each new joint position and the prediction of the new point based on existing sequence of each human is calculated. In row 10 these errors are then summed for each group of new joint positions and the humans in the scene. Next, these summed-up errors for two humans and two new data groups can be visualized in an error matrix such as

$$E_{H1,H2} = \begin{bmatrix} e_{JG1 \rightarrow H1} & e_{JG1 \rightarrow H2} \\ e_{JG2 \rightarrow H1} & e_{JG2 \rightarrow H2} \end{bmatrix}, \quad (2)$$

where $e_{JG1,H1}$ is the value of the summed up errors between the new joint position group 1 and Human $H1$, $e_{JG1 \rightarrow H2}$ for joint position group 1 and Human $H2$ and so forth. In row 13-16 it is then assessed which trace of all permutations of the rows of $E_{H1,H2}$ results in the lowest total error, such as

$$T_{b,p} = \operatorname{argmin}(\operatorname{tr}(P(E_{H1,H2}))), \quad (3)$$

where $T_{b,p}$ is the trace of the best permuted error matrix E , $P(\cdot)$ is the set of all permutations, $\operatorname{tr}(\cdot)$ is the trace of a matrix and $E_{H1,H2}$ is the error matrix of two humans. The corresponding permutation indicates if the joint position groups need to be swapped or if they are assigned to the sequence of the correct human. For two permutations this can be easily computed, but the usage of permutations of the error matrix is also scalable to evaluating situations with more humans present. For a scene where i humans are tracked, the error matrix would be a $i \times i$ -matrix with the factorial of i permutations.

Algorithm 2 Associate New Data to Correct Human

Require: joint position groups: $[x, y, z]$ of 6 joints per Human

- 1: **for** each human **do**
 - 2: **for** each joint position group **do**
 - 3: **for** each joint **do**
 - 4: Make prediction of new joint position based on moving average of past joint positions (Eq. 1)
 - 5: **if** first frame **then**
 - 6: Take initial position as prediction value
 - 7: **else if** prediction invalid **then**
 - 8: Take last known valid joint position as prediction value
 - 9: **end if**
 - 10: Calculate error new joint and prediction
 - 11: **end for**
 - 12: Sum error values per joint position group and human combination
 - 13: **end for**
 - 14: **end for**
 - 15: **for** each permutation of error matrix **do**
 - 16: Evaluate trace of permuted error matrix (Eq. 2)
 - 17: **end for**
 - 18: Best situation is permutation with lowest sum of errors (Eq. 3)
 - 19: **return** Joint position groups reordered based on best situation
-

2) *Resolving Occlusion:* After the new data points are assigned to the correct sequence, there is looked at the detection of occlusions and how to relocate those points so they better represent the original real life location. Occlusions can for example occur when a person is facing the robot with its right side. The left hip, elbow and shoulder are in this case blocked from sight by the rest of the body of the human. However, in most cases OpenPose still recognizes these joints and thus provides a pixel coordinate point, but when these pixel coordinates are converted to a 3D position, the found position often is closer to the joint facing the camera than the actual (obstructed) location. This phenomenon is shown in Figure 5 for all six joints. It is clear that the left joint locations from OpenPose are not close to the original location of the joint. Algorithm 3 is thus proposed to detect if a joint position is misplaced due to occlusion. When this is the case, Algorithm 4 is used to estimate the original position of an occluded joint based on the distance between the joint facing the camera and the misplaced joint.

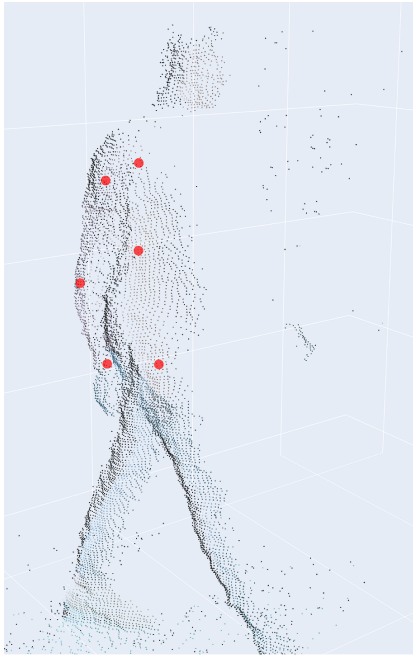


Figure 5: An example of wrongly estimated joint positions due to occlusion.

For the detection of occlusion it is first checked in row 2-4 if one of the joints of the current joint pair is missing. If both positions have a valid position, it is checked in row 5-8 if one of both positions lays too close to the borders of the field of view. If this is the case, the human is only partly visible and thus these missing joint positions should not be repositioned inside the field of view. Lastly, in row 9-24 the norm of the new joint pair is compared to the average of all previous correct norms of this joint pair for each human. If the difference is too small or too big, it needs to be determined which of the two joints is correct and which one has to be substituted. The limits on the differences are experimentally determined. If the difference is smaller than the average minus 0.05m, the other joint pairs in the new joint position group are evaluated to determine which joint positions are closer to the robot and thus which side of the human is facing the robot. If the other joint pairs are also not valid, the point with the lowest confidence from OpenPose is substituted. Alternatively, if the difference in norms is higher than the average plus 0.20m, this means that one of the found joint positions is positioned somewhere in the background and not on the body of the human. In this case also the point with the lowest confidence from OpenPose is substituted.

Algorithm 3 Resolve Occlusions

Require: joint positions of Human: $[x, y, z]$ of 6 joints in 3 pairs

```

1: for each joint pair do
2:   if one of joint positions invalid then
3:     Continue with next joint pair
4:   end if

5:   Calculate perpendicular distances joints to closest
   border field of view
6:   if one of the distances  $\leq 0.1m$  then
7:     Continue with next joint pair
8:   end if

9:   Calculate norm of distance between joints of joint
   pair
10:  if norm joint pair is within average limits then
11:    Add norm of joint pair to average
12:  else if norm joint pair is too small then
13:    for each other joint pair do
14:      Calculate norm of distance between joints of
      other joint pair
15:      if norm other joint is within average limits
16:      then
17:        Identify if the Human is facing the robot
        with its right or left side
18:        Substitute joint corresponding to other
        side with FJI algorithm
19:      end if
20:    end for
21:  else if norm joint pair is too big then
22:    Identify joint that is furthest away from robot
    location
23:    Substitute furthest joint with FJI algorithm
24:  end if
25: end for

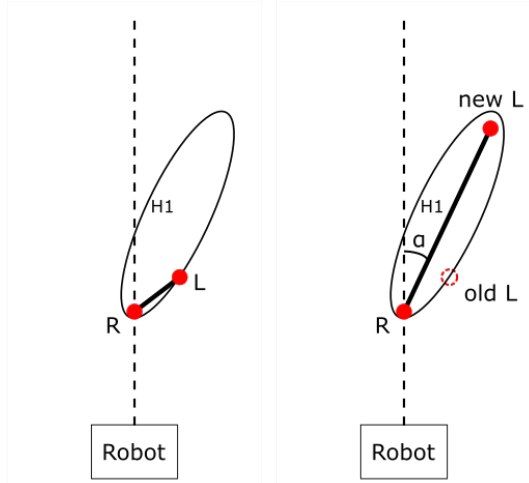
```

The Far Joint Inferring (FJI)-algorithm of Algorithm 4 is used for the substitution of a joint position if the norm of the accompanying joint pair is smaller than the threshold. First, the 2D vector from the robot towards the closest joint position is calculated in row 1. Then in row 2 a new vector is created with the same direction and the average measured norm of this joint pair. This vector starts in the closest joint position and is rotated based on the ratio between the old norm and the average norm of the joint pair. This ratio is then multiplied with 45° in row 4 to determine the angle of rotation. This method is based on the fact that the furthest joint position is found closer to the closest joint position if the human is oriented perpendicular with respect to the robot. A threshold of 45° is chosen because if a human would turn further, the robot would again be able to see the furthest joint directly and the original norm thus would measure within limits. Lastly in row 5 the newly found joint is substituted into the joint position group. This process is schematically represented in Figure 6 for the left joint position.

Algorithm 4 Far Joint Inferring algorithm

Require: joint pair: $[x, y, z]$ of 2 joints in joint pair
Require: furthest point, closest point

- 1: Obtain 2D vector from robot to closest joint position
 - 2: Create new vector with this direction and length of average measured norm of joint pair
 - 3: angle correction $a_c = \text{norm}/\text{average norm} * 45^\circ$
 - 4: Rotate adjusted norm around Z -axis of closest joint with a_c° towards side of old point
 - 5: Substitute old furthest joint position with newly acquired joint position
-



(a) Joint positions of shoulders as received from OpenPose. The distance between R and L is the old norm.
 (b) Joint positions of shoulders after application of FJI algorithm. The distance between R and the new point L is the average norm of a particular joint pair.

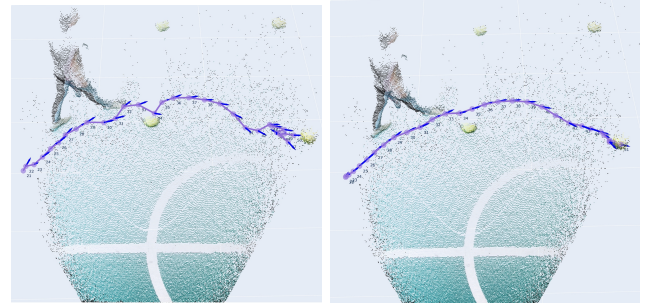
Figure 6: Top-down schematic representation of the repositioning of an occluded joint, where R is the right and closest joint position, L is the left and furthest joint position, α is the rotation angle and the dotted line represent the vector from the robot towards the closest joint position.

3) *Kalman Filter for Trajectory Smoothing:* With the occluded joint locations repositioned, the 2D location of each human is determined by taking the average of all new joint locations. The orientation is calculated by taking a weighted moving average of the past five velocity vectors, as illustrated in

$$\begin{bmatrix} v_{x,t+1} \\ v_{y,t+1} \end{bmatrix} = \frac{1}{\sum_{n=0}^{N-1} 2^{N-n}} \sum_{n=0}^{N-1} 2^{N-n} \cdot \left(\begin{bmatrix} x_{t-n} \\ y_{t-n} \end{bmatrix} - \begin{bmatrix} x_{t-n-1} \\ y_{t-n-1} \end{bmatrix} \right), \quad (4)$$

where $v_{x,t+1}$ and $v_{y,t+1}$ denote the x - and y -component of prediction for the velocity vector, x_t and y_t are the x - and

y -position at frame t , N is the total amount of frames that is looked back, n is the amount of frames and t is the current frame number. The total resulting vector of the form $[x_t, y_t, \theta_t]$ is subsequently used as input for a Kalman filter with a constant velocity model for each human. With a Kalman filter new measurements are compared to the beliefs of an underlying model and are updated based on the likelihood of observing the new measurements. The result in this case is the smoothing of the trajectory by repositioning the new data point based on the assumption of a constant velocity model and Gaussian distributed errors. The effect of applying a Kalman filter on an example sequence is shown in Figure 7. It can clearly be seen that the resulting trajectory with a constant velocity Kalman filter applied is a lot smoother and thus better represents a casually walking human person. Lastly, the filtered position and orientation are added to the time sequence of each human.



(a) Example sequence without a Kalman filter applied. (b) Example sequence with a constant velocity Kalman filter applied.

Figure 7: An example sequence to highlight the effect of applying a constant velocity Kalman filter to each new joint position before adding it to the sequence.

C. Intention Prediction

In the last part of the pipeline from Figure 2 the obtained position and orientation of each human is used to predict the most likely goal of each human. In Subsection III-C1 it is explained how the likelihood of each goal hypothesis is calculated using a 2D Gaussian function for the probability distribution of the positions and a custom function for the probability distribution of the orientations. This custom function is derived from the work of [15] but expanded to work in an open field instead of a hallway environment. Next, in Subsection III-C2 the method for sharing data between the two robots is described.

1) *Likelihood Calculation:* The intention prediction of each human is done by evaluating the most likely semantic goal based on Bayesian mathematics. During the experiments there are four goals to be evaluated: goal b_1 , goal b_2 , goal b_3 and goal b_4 . Each hypothesis H_i represents that the most likely goal of the human is goal b_i . The hypothesis with the highest likelihood is deemed to most likely goal of that human at a

particular time frame. The probabilistic model for each of the four goal hypotheses is defined as

$$p_i(H_i|x_i, y_i, \theta_i) = p_i(H_i|\theta_i) * p_i(H_i|x_i, y_i), \quad (5)$$

where it $p_i(H_i|x_i, y_i, \theta_i)$ is the total combined likelihood of hypothesis H_i , $p_i(H_i|x_i, y_i)$ is the likelihood for H_i given the position $[x_i, y_i]$ of the human and $p_i(H_i|\theta_i)$ is the likelihood for H_i given the orientation θ_i of the human. These likelihoods can be multiplied because it is assumed they are independent observations. Each of these likelihoods can be expressed using Bayes' rule as

$$p_i(H_i|obs) = \frac{p_i(obs|H_i)p_i(H_i)}{p_i(obs)}, \quad (6)$$

where obs can be either $[x_i, y_i]$ or $theta_i$, $p_i(H_i|obs)$ is the likelihood of hypothesis H_i given a certain observation, $p_i(obs|H_i)$ the probability of observing obs given hypothesis H_i is true, $p_i(H_i)$ the probability of the system beginning in state H_i and $p_i(obs)$ the likelihood of seeing observation obs . While there is a finite set of hypotheses, the law of total probability is applied such that

$$p_i(H_i|obs) = \frac{p_i(obs|H_i)p_i(H_i)}{\sum p_i(obs|H_i)p_i(H_i)}. \quad (7)$$

When Eq. (5) and (7) are combined, the prior knowledge terms cancel each other and the equation for the total likelihood becomes

$$p_i(H_i|x_i, y_i, \theta_i) = \frac{p_i(x_i, y_i|H_i)}{\sum p_i(x_i, y_i|H_i)} * \frac{p_i(\theta_i|H_i)}{\sum p_i(\theta_i|H_i)}. \quad (8)$$

The probabilities are automatically normalized because of the division by the sum of all probabilities, so the resulting likelihoods will always add to 1.0.

Next, the probability density function for the position of a human is defined as a homogeneous 2D Gaussian function with the mean $[x_i, y_i]$ at the location of goal i . The 2D Gaussian is chosen because it increases gradually to the mean, resulting in an increase in likelihood when a human approaches a certain goal from any direction. The 2D Gaussian function is expressed as

$$p_i(x_i, y_i|H_i) = A * \exp\left(-\left(\frac{(x-x_i)^2}{2\sigma^2} + \frac{(y-y_i)^2}{2\sigma^2}\right)\right) \quad (9)$$

where $[x, y]$ is the position of the human, $[x_i, y_i]$ the location of goal i , the amplitude A and the standard deviation σ .

As the probability density function for the orientation of the human a custom function is chosen based on the work of [15]. The custom function consists of the positive part of the inner product between the corresponding vector of the human orientation θ and the vector from the human towards goal b_i , multiplied by a constraint parameter f_c . This ensures the human has to be oriented in the direction of the goal to have a probability higher than zero. The equation is given by

$$p_i(\theta_i|H_i) = \max(0, \frac{\Theta_i \cdot d_{H_i}}{|\Theta_i||d_{H_i}|})f_c, \quad (10)$$

where $\Theta_i = \begin{bmatrix} \sin \theta_i \\ \cos \theta_i \end{bmatrix}$ is the orientation vector with a length $|\Theta_i| = 1$, d_{H_i} the 2D vector from the human towards goal b_i with length $|d_{H_i}|$ and lastly the constraint parameter f_c . This parameter has a value between 0 and 1 and it decreases the influence of the orientation on the probability when a goal is not the closest goal of the human. The equation is given by

$$f_{C,i} = \frac{\min(|d_{H_1}|, |d_{H_2}|, |d_{H_3}|, |d_{H_4}|)}{|d_{H_i}|}. \quad (11)$$

Lastly, a fifth hypothesis H_{NT} is added to account for the uncertainty in the model. The value for this hypothesis is chosen to be constant, so it has a relatively high likelihood if the likelihoods of the other hypotheses are low and vice versa due to the normalization in Eq. (8). The value of this constant remains a design parameter and behaves like a threshold value for when then likelihood of a hypothesis before normalization starts to count.

2) *Data Sharing*: The previously explained parts of the pipeline all relate to the data conversion in a single robot, but in this subsection it is explained how the hypotheses of each robot are merged with the hypotheses of the other robot. The one of the main goals of this thesis is to identify improvements in human intention prediction due to data sharing, so a simple data sharing method is chosen for comparison to the individual cases. First, the frames per second of both robots need to be measured. Then, the sequences are compared at the timestamp of the slowest robot. At that time frame, the values from most recent time frame of the other robot are taken. This conversion is expressed as

$$f_{n,R2} = \text{floor}(f_{n,R1} * \frac{dt_{R1}}{dt_{R2}}), \quad (12)$$

where $f_{n,R2}$ is the frame number of Robot $R2$, $f_{n,R1}$ the frame number of the slowest Robot $R1$, dt_{R1} the time between frames for Robot $R1$ and dt_{R2} the time between frames for Robot $R2$. The time between frames can also be defined as the $\frac{1}{fps}$ where fps denotes the frames per second of a robot camera. Once two probability distributions are selected, the likelihoods of the combined goal hypotheses are calculated by adding the individual hypotheses element-wise following

$$p_{i,R1+R2}(H_i|v_{i,R1}, v_{i,R2}) = \frac{p_{i,R1}(H_i|v_{i,R1}) + p_{i,R2}(H_i|v_{i,R2})}{\sum_{R1,R2} p_i(H_i|v_i)} \quad (13)$$

where $p_{i,R1+R2}(H_i|v_{i,R1}, v_{i,R2})$ is the likelihood after data sharing, $p_{i,R1}(H_i|v_{i,R1})$ is the combined likelihood of Robot $R1$, $p_{i,R2}(H_i|v_{i,R2})$ is the likelihood of Robot $R2$, $v_{i,R1} = [x_i, y_i, \theta_i]_{R1}$ is the position and orientation vector of Robot $R1$, $v_{i,R2} = [x_i, y_i, \theta_i]_{R2}$ is the position and orientation vector

of Robot $R2$ and $\sum_{R1,R2} p_i(H_i|v_i)$ is the sum of likelihoods of both distributions. The division by the sum of likelihoods is done to ensure the likelihoods sum to 1.0. The No Target hypothesis H_{NT} remains constant such that

$$p_{i,R1+R2}(H_{NT}) = p_{i,R1}(H_{NT}) = p_{i,R2}(H_{NT}) = c_{NT}. \quad (14)$$

Algorithm 5 Goal Estimation Pipeline

1: **for** every new frame of robot with slowest fps **do**
2: **for** each robot **do**

Phase 1 – Perception

3: Obtain depth and color frame from Kinect
4: Analysis color frame by OpenPose
5: Transform pixel values of joint positions into 3D coordinates (Alg. 1)

Phase 2 – Trajectory

6: Associate each group of joint positions to correct Human (Alg. 2)
7: Detect and substitute occluded joints (Alg. 4)
8: Take average (x, y) -position
9: Calculate 5 frame moving average velocity direction as orientation (Eq. 4)
10: Apply constant velocity Kalman filter to smooth final position and orientation (Eq. 7)
11: Add position and orientation to time sequence

Phase 3 – Intention Prediction

12: Calculate likelihood for all goal hypotheses with 2D-Gaussian (Eq. 9) and orientation towards the goals (Eq. 10)
13: Add No Target hypothesis and normalize individual likelihoods (Eq. 8)
14: Add individual likelihoods to time sequence
15: **end for**

Phase 4 – Data Sharing

16: Find most recent time stamp for both robots (Eq. 12)
17: Combine likelihoods (Eq. 13)
18: Add No Target hypothesis and normalize combined likelihoods (Eq. 13, Eq. 14)
19: Add combined likelihoods to time sequence
20: **end for**

D. Summary

In this subsection a summary is given of the complete Goal Estimation Pipeline discussed in this section. In Algorithm 5 an overview of all steps is shown corresponding to Figure 2. First, the color frame and depth frame are retrieved from the Kinect at a time t (row 3). The color frame is analysed by the OpenPose algorithm and the joint positions of three joint pairs are extracted: the shoulder-, elbow- and hip-joints (row 4). By using the depth frame from the Kinect, the 3D joint positions

of each followed joint are determined (row 5). To overcome mismatches due to the compression inside the Kinect, the search area around the point supplied by OpenPose can be expanded. After that, the newly found points are assigned to the correct sequence based on the error with a moving average prediction of each joint position (row 6) and occluded points are repositioned with the Far Joint Inferring algorithm (row 7). The average position of each human is calculated as the 2D average of each joint position and the orientation of each human is determined by calculating the five frame moving average of the velocity vector (row 8-9). Then a Kalman filter is used to reposition the trajectory assuming a constant velocity dynamic model (row 10). The resulting vector $[x_t, y_t, \theta_t]$ for time frame t is saved for each robot (row 11). Next, the likelihoods are evaluated for each goal hypothesis and saved for time frame t (row 12-13). Lastly, the closest time stamps of both robots are evaluated and the likelihoods are combined by adding the likelihood of each goal hypothesis element-wise (row 15 – 16). The result is normalized so the sum of all likelihoods remains 1.0 and the results are saved in the sequence for time frame t (row 17-18). The output is thus a position and orientation sequence for every time frame t with the accompanying values for five hypotheses for each robot individually and for the situation where data is shared.

IV. EXPERIMENTS

In this section it is explained how the experiments were performed. As highlighted in Section II, the experiment setup consists of two robots, two humans and four goals. The robots are represented by two Kinect V2 camera's connected to two laptops running Ubuntu 20.04. On each laptop a custom Python script is executed that retrieves the color- and depth-frame from the Kinect camera. The frames per second measured is highly influenced by the processing speed of the script, so the script is designed in such a way that the conversion from the single depth value towards a $[x, y, z]$ -position using the focal length of the camera is done after the recording of the video. This results in an average frames per second of 5Hz for robot $R1$ and 6 Hz for robot $R2$. The used script can be found in the repository² accompanying this thesis, of which the file structure is described in Appendix C. Next, the four goals are represented by four soccer balls. All balls have the same color yellow, which is distinct from the green field so they can be clearly recognized in each frame. The balls are placed in four different configurations named A-D. Each configuration also has a different set of possible starting positions for the humans named $a-m$, of which an overview is displayed in Appendix B. An example of an experiment setup is shown in Figure 8. The experiments are conducted in series of ten trials per two participants. The trials are ordered in such a way that the same exact experiment is not repeated by the same two participants inside a series. Furthermore, in a series of ten trials subsequent starting positions of the humans as well as starting positions of the goals are always varied.

²https://gitlab.tue.nl/et_projects/svintentionrecognition

This is done to ensure more random behavior and to eliminate external influences on the decisions of the participants as much as possible. All experimental configurations can be found in the repository accompanying this thesis.

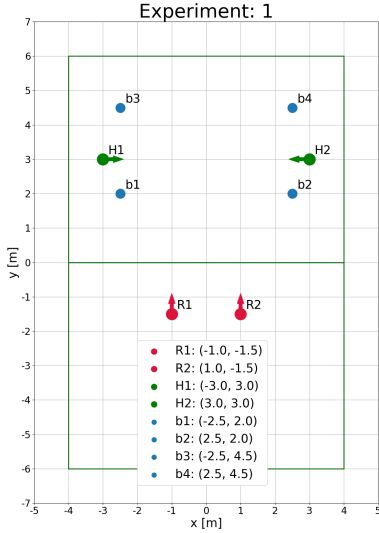


Figure 8: The starting positions of all humans and goals in Experiment 1. The balls are placed in Configuration D, while the humans start in Configuration *a*.

The experiments were performed spanning three days with the help of 19 different voluntary participants. Each participant has signed an informed consent form, of which a template can be found in Appendix E. Due to two participants participating multiple days, a total of 110 experiments were performed. The data of 12 experiments was corrupted due to saving issues at the used laptops and thus unusable, so the final results are 98 correctly captured experiments. Each trial resulted in two videos with a length of 15 seconds, consisting of 100-120 frames each depending on the frame rate of the camera. All gathered data can also be found in the repository³ of this thesis.

V. RESULTS

In this section the effect of the application of the Goal Estimation Pipeline on the performance within the experiments is discussed. First, the *Perception* part of the pipeline is evaluated. This first part is responsible for extracting the 3D joint locations from OpenPose data, as explained in Section III-A. Next, the performance of the second part *Trajectory* is determined by identifying the main causes for incomplete or insufficient 2D time sequences. After that, the performance of the human intention prediction as explained in Section III-C is evaluated based on the amount of correctly identified goals and the percentage of the time and absolute distance at which the correct hypothesis first becomes dominant. The results of the individual situations are compared to the results

³ https://gitlab.tue.nl/et_projects/svintentionrecognition

when data is shared between robots. Specific examples from the experiments are used to highlight the improvements with respect to the latter case.

A. 3D Joint Position Estimation

In this section the performance of *Perception* part of the Goal Estimation Pipeline explained in Subsection III-A is evaluated. This is done by determining in how many cases Algorithm 1 returns a valid joint position after a maximum of 8 iterations. In table I the cumulative amount of valid joint positions after each amount of iteration i is shown. Also the percentage of the total amount of assessed joint positions is shown. It can be derived from this table that in 80.29% of the cases a valid joint position is found with the first time the algorithm is executed (the initial value of i is 2). This leaves about 20% of cases where a substantial mismatch between the pointcloud and the color frame due to the compression in the Kinect is observed. The algorithm is automatically terminated after 8 iterations, resulting in the fact that in 93.94% of the evaluations eventually a valid joint position is found. For the remaining 6.06% the compression resulted in such a mismatch that a valid point could not be found with the closest 256 pointcloud values.

Table I: Cumulative amount and percentage of found valid joint positions after amount of iterations i of Algorithm 1.

Amount of Iterations:	$i = 2$	$i = 3$	$i = 4$	$i = 5$	$i = 6$	$i = 7$	$i = 8$	Total
Amount [-]	80408	86424	89360	91213	92502	93429	94068	100146
Percentage [%]	80.29	86.30	89.23	91.08	92.37	93.30	93.94	100.00

B. Construction of 2D Human Position and Orientation Sequences

In this section the performance of the *Trajectory* part of the Goal Estimation Pipeline as shown in Figure 2 is evaluated. In this part multiple joint positions in a 3D coordinate system are converted to one position and orientation of a human at each time frame. These positions and orientations are then filtered by a Kalman filter and chained together to form a time sequence of the trajectory of the human. When a human reaches a goal, that goal is marked as reached based on automatic detection by position proximity. In Table II the amount of sequences for each human-robot pair of which is detected sequences end at the correct goal. Some of these goals could never be detected correctly (for example because the goal is outside the field of view of the robot), others are wrongly detected due to gaps in the data or other inconsistencies. In this section the identified causes for not achieving a perfect amount of correctly detected end goals are discussed. Then in the next section the prediction of the correct goals are discussed and the individual cases are compared to the case where data is shared.

Table II: The amount of correctly assessed sequences from each robot for each human in the scene - **Goal: Balls**

Amount Correct	R1	R2
H1	63 of 98	40 of 98
H2	49 of 98	60 of 98

Gap in Data Sequence when Reaching Goal

In Figure 9 the position and orientation sequences of both humans are displayed. It is visible that the sequence of Human *H2* nicely begins at the border of the field of view and ends at one of the goals. However, in the sequence for Human *H1* a data gap is present exactly when the human is reaching the goal. Due to this gap in the data, the goal is not marked as reached and the sequence is not separated in the two sequential tasks: walking towards one of the goals and walking towards one of the robots. A possible explanation for this data gap is that the infrared depth rays from the Kinect data become more sparsely distributed if they have to travel further. This results in a more coarse distribution of data points in the back of the field, which can result in the case that there is no matching depth point for the joint location as determined by OpenPose.

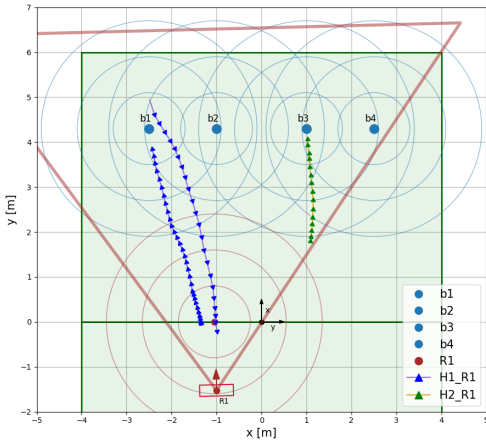


Figure 9: Example of a situation where there is a gap in the data sequence exactly when a human reaches the goal. - **Experiment 77, R1**

Reaching Goal Outside Field of View

In Figure 10 the two sequences from Robot *R2* in Experiment 62 are displayed. It is shown that Human *H2* is moving towards goal *b1*, but it is not detected as reached because the goal is positioned outside the field of view of the robot. This is the reason the sequence is not split and the second part of the sequences from that goal to Robot *R2* is also visible. These mislabeling errors are inherent to the chosen configurations of the positions of the goals.

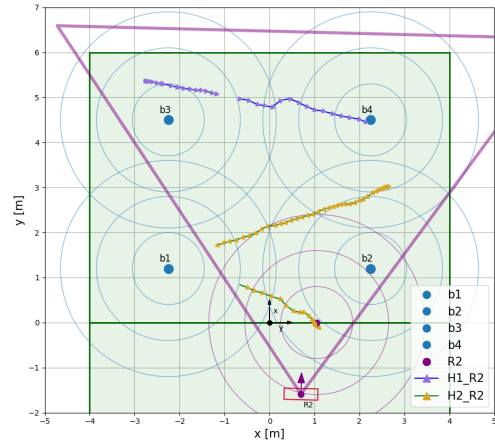


Figure 10: Example of a situation where the goal is outside the field of view of the robot. - **Experiment 62, R2**

Location Offset of Goals

In Figure 11 the sequences for both humans from Robot *R1* are displayed. The trajectory shows that Human *H1* is walking towards goal *b3* and the remainder of its walking pattern indicates that he has reached it. However, the location of the bend in the trajectory does not correspond with the predefined location of the goal. The same holds for Human *H2* and goal *b4*. The result is that the goal is not marked as reached. This can be caused by the position of the goals during the experiments not being at exactly the predefined (x, y) -location in the field. Alternatively, it can be caused by an offset in the orientation of the camera or a propagated offset by the Kalman filter smoothing.

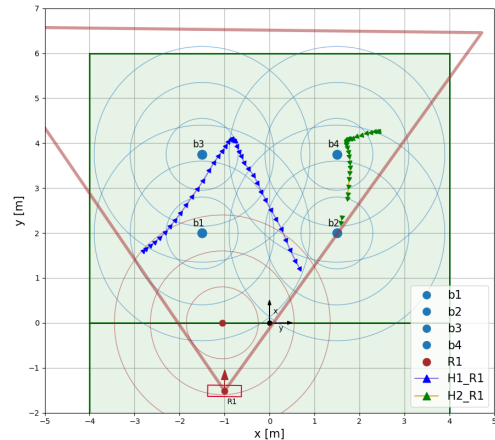


Figure 11: Example of a situation where there is a substantial location offset in the goals and/or the sequence. - **Experiment 43, R1**

Insufficient Data due to Occlusions

In Figure 12 the data sequences of both humans from Robot *R1* are shown. The sequence of Human *H1* is smooth and continuous, while the sequence belonging to Human *H2* is divided into three parts. This is caused by the fact that Human *H1* is crossing in front of Human *H2* with respect to Robot

$R1$, such that Human $H2$ is only partially visible during some time. The absence of data from Human $H2$ results in a high uncertainty in the update step of the Kalman filter, so when data is available again the positions are corrected wrongly. Furthermore, the moving average of the velocity vector also becomes increasingly unreliable with a non-continuous data sequence, resulting in orientations that can point in the wrong direction.

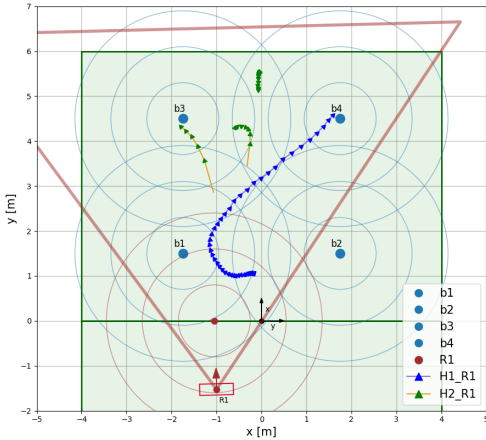


Figure 12: Example of a situation with insufficient OpenPose data for $H2$. - **Experiment 83, R1**

Incorrect Goal Detected as Reached

In Figure 13 the sequences of both humans from Robot $R2$ are displayed. At first glance both sequences are looking fine, but the sequence of Human $H2$ terminates at goal $b4$, while the person is walking past this goal towards goal $b3$. This goal is thus mislabelled by the position proximity method used to check if a goal is reached by a human. The trade-off between sensitivity of the the proximity method and mislabelling of humans walking past goals will always remain, but is mitigated by choosing the detection range at the current experimentally obtained value of 0.25m.

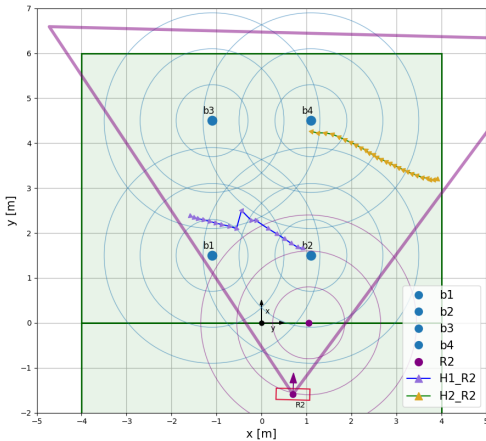


Figure 13: Example of a situation where a human ($H2$) walks too close to another ball such that it is mislabeled as reached. - **Experiment 38, R2**

Sequences of Humans Flipped

In this subsection three causes are highlighted for the unintentional swapping of sequences. In Figure 14 both sequences belonging to Experiment 94 are displayed. It is visible that the the trajectory for Human $H1$ from Robot $R1$ starts at the left side of the middle line, but then suddenly transforms into the sequence of Human $H2$. The path of Human $H1$ shows a gap and continues after a few frames in the middle of the field towards goal $b2$. A similar trajectory is visible from Robot $R2$ but it belongs to the other human. The sequences of each robot for both humans are thus swapped with respect to each other. In this case it occurs because Human $H1$ is tracked while he is outside the area and on entering the field the evaluation of the data association algorithm becomes very low for both cases, resulting in a wrong assessment that is then propagated throughout the analysis.

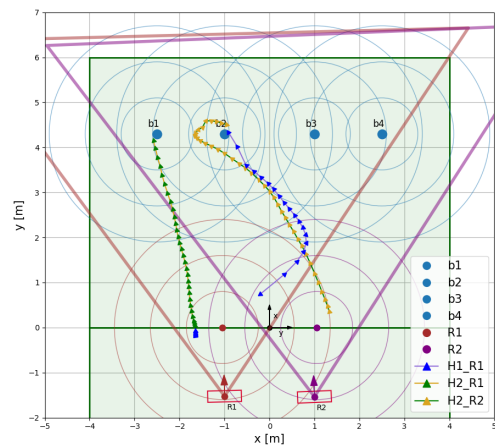


Figure 14: Example of a situation where the sequences of both humans are flipped from a few frames after the start. - **Experiment 94**

In Figure 15 both sequences of Experiment 26 are shown. The path of Human $H1$ by Robot $R1$ is cut off at goal $b1$, while the eventual goal is goal $b4$. The remainder of the sequence is then assigned to Human $H2$ because the actual path of Human $H2$ (as indicated by the estimation of robot $R2$) crosses in front of the path of Human $H2$ with respect to robot $R1$. Robot $R1$ thus loses track of human $H1$ for a few seconds, after which the sequences are swapped for the remainder of the experiment.

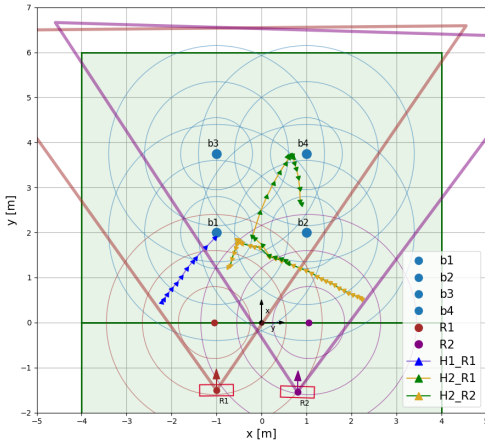


Figure 15: Example of a situation where the sequences of both humans are flipped after a crossing. - **Experiment 26**

Lastly, in Figure 16 all sequences of Experiment 21 are displayed. In this case the sequence for Human $H1$ of Robot $R1$ is swapped with respect to the Robot $R2$ when Human $H2$ exits the field of view of Robot $R1$. Robot $R2$ still sees the human continuing his path towards goal $b2$, but for Robot $R1$ the newly observed positions from that point are assigned to the wrong human sequence. This can occur when some values in the error matrix $E_{H1,H2}$ from Eq. (2) are invalid, which can result in a wrong assignment of a frame which is then propagated to all further frames.

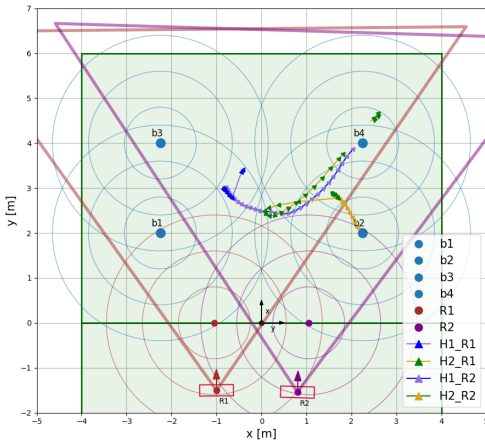


Figure 16: Example of a situation where the sequences of both humans are flipped after a human exits the field of view of one robot. - **Experiment 21**

C. Evaluation of Data Sharing

In this section the results obtained from implementing the third part of the Goal Estimation Pipeline *Intention Prediction* (Subsection III-C) on the data sequences. For this last part the sequences of the 2D position and orientation of both humans are used to calculate the likelihoods for which goals the humans are most likely to go to over time. These probability distributions are made for both robots individually and for the situation where a simple form of data sharing is

present between the two robots. The parameter values for the intention prediction function are experimentally determined to be adequate and are consequently chosen as follows: The amplitude and standard deviation of the 2D Gaussian of Eq. (9) are $A = 2.0$ and $\sigma = 0.8$ respectively. The constant value for hypothesis H_{NT} is chosen equal to $c_{NT} = 0.01$. The probability distributions are compared based on three metrics:

- 1) C_h : the correctness of the hypothesis as a boolean.
- 2) $T\%$: the percentage of the total time at which the correct hypothesis first becomes dominant.
- 3) d_G : the absolute distance in meters m from the human to the goal at the time the correct hypothesis first becomes dominant.

Naturally, $T\%$ and d_G do not have a value in the case that the final hypothesis is not the correct hypothesis, and these experiments are thus excluded from the resulting average values for $T\%$ and d_G . The metrics were chosen because they reflect some important values of a real-life situation well. The most important value of course is that the final goal is estimated correctly. A wrong assessment of the situation may later result in unexpected behavior. If the goal is estimated correctly, the percentage of the total time at which the correct hypothesis first is dominant (measured from the first time the human is detected by the robot until the goal is reached) indicates the speed of convergence of the used intention prediction algorithm. The absolute distance from the human towards the goal at the time the correct hypothesis first becomes dominant indicates the real world relevance of the intention prediction algorithm. The aim of intention prediction is to estimate the goal of the agent before this goal is reached. Further away from the goal generally indicates a better working prediction algorithm, because the possible anticipation time is increased.

Overall Analysis

Firstly, there is looked at the first part of all sequences where the humans have one of the balls as their goal. The amount of sequences (two sequences per experiment) where the final goal is estimated correctly is shown in table III. Several reasons for why not all sequences are correct have already been elaborated on in Section V-B, but the main point of interest is the increase in the amount of sequences that are estimated correctly when data is shared. In the situation where data is shared for 77% of the sequences the goal of the human is estimated correctly, while it is only 57% for R1 and 51% for R2 individually. On average, there is thus an increase in correctly estimated goals of approximately 43.40% when data is shared.

In Table IV averages of the other two metrics are displayed. It is shown here that the sharing of data on average results in a decrease in $T\%$ of 7% with respect to robot R1 and no decrease with respect to robot R2. For d_G an increase of 13% is seen when compared to only robot R1 and an increase of 10% when compared to only robot R2. It is thus clear that even a rather simple form of data sharing results in an

Table III: The amount of correctly assessed sequences C_h in all experiments - **Goal: Balls**

Config	No Data Sharing		Data Sharing
	R1	R2	R1+R2
	C_h [-]	C_h [-]	C_h [-]
all	112 of 196	100 of 196	152 of 196

Table IV: The mean over all experiments of the percentage of total time $T\%$ and absolute distance d_G at which a correct assessment is made - **Goal: Balls**

Config	No Data Sharing				Data Sharing	
	R1		R2		R1+R2	
	$T\%$ [%]	d_G [m]	$T\%$ [%]	d_G [m]	$T\%$ [%]	d_G [m]
all	52	1.60	49	1.64	49	1.81

improvement of most metrics. It is interesting to look at for which situations sharing of data results in the highest gains and if there are situations in which the performance is less good or even worse when data is shared.

A first check that is made is to cluster all sequences based on which human is tracked and compare which robot tracks which human better. These results are shown in Table V. It is to be expected that R1 would have more correct assessments of H1, because H1 always starts on the left side of the field where R1 is located. The same holds for H2 and R2. In Table V it is shown that this is indeed the case. If R1 tracks H1 more and R2 tracks H2 more, it would also be expected if the data is shared that this effect is mitigated because both robots together would oversee the whole field. This is also noticeable in Table V as the amount correct of both humans for the data-sharing case is almost equal.

Table V: The amount of correctly assessed sequences C_h clustered based on the two humans present in the experiments - **Goal: Balls**

Human Name	No Data Sharing		Data Sharing
	R1	R2	R1+R2
	C_h [-]	C_h [-]	C_h [-]
H1	63 of 98	40 of 98	77 of 98
H2	49 of 98	60 of 98	75 of 98

Crossings

Another interesting clustering of data is displayed in Table VI. Here a distinction is made between sequences that are part of a crossing and sequences that do not cross with the other sequence in the situation. The most noticeable correlation in this table is that the effect of data sharing has a bigger positive influence on the metrics in situations where a crossing is present, namely on average an increase of 9% for $T\%$ against a decrease of 7% for the non-crossing sequences and an increase of 19% in d_G against an increase of just 4% in the non-crossing sequences. It is thus concluded that the vast majority

Table VI: Experiment results for $T\%$ and d_G , clustered based on if a crossing of trajectories happened in that particular experiment - **Goal: Balls**

Crossing in experiment	No Data Sharing				Data Sharing	
	R1		R2		R1+R2	
	$T\%$ [%]	d_G [m]	$T\%$ [%]	d_G [m]	$T\%$ [%]	d_G [m]
True	61	1.57	61	1.59	56	1.88
False	43	1.63	35	1.69	42	1.74

of gained improvement from sharing data in earlier detection of the correct goal is to be found in situations where a crossing is present.

Configurations of Ball Starting Positions

Next the data is clustered based on the different starting positions of the soccer balls. As discussed in Section II, four different configurations of starting positions are used in the experiments: Configuration A corresponds to the four balls being placed in a 1x1-meter square in the middle of the field, Configuration B correspond to a 2x2-meter square, configuration C corresponds to the balls being placed in a line 1.5 meter from the goal line and configuration D corresponds to the balls being placed towards the four corners of the field. The results of the amount of correctly evaluated sequences are shown in Table VII. It is shown that the sharing of data results in the lowest increase in C_g in configuration A, namely an average increase of 19% for H1 and 24% for H2. Furthermore, neither of the totals of the data sharing situation exceed the highest value of their respective individual robots, thus indicating that no additional goals have been estimated correctly by sharing data.

Next, both configuration B and C have a moderate increase in the amount of correctly estimated goals, namely an average increase in configuration B of 50% for H1 and an increase of 30% for H2. In configuration C an average increase of 68% for H1 and in increase of 27% for H2 is noted. The highest increase in correctly estimated goals when data is shared is however in configuration D, namely an average increase of 88% for H1 and an increase of 83% for H2. Next to that, the total amount of correctly estimated goals as a percentage of the total is highest in configuration B and D where the balls are placed further away from each other.

It is thus derived from the data that the sharing of data is more prevalent in situations where the goals are more sparsely located. With configurations A and C less sparsity leads to more mislabeling of goals, which results in a lower total correct count in these situations. With the more sparser configurations B and D, all goals may not be in the field of view of both robots. The sharing of data thus has an immediate gain on the amount of correctly estimated sequences by adding information that would not have been visible for the other robot individually. To further quantify the gains of sharing data in each configuration of ball starting positions, the increases in $T\%$ and d_G are displayed in Table VIII. The increases are displayed percentage-wise with respect to the data sharing

case. The absolute averages per situation can be found in Appendix Table D.8. All configurations denote an average positive increase in the distance to the goal, with Configurations B leading with an average increase of 32%, which is about 0.59m earlier from the ball. Situations with Configuration D have the second best improvement with an average increase of 13% and situations with Configuration A and C follow with an increase of 7% and 8% respectively. In Configuration A and C also an increase in $T_{\%}$ of respectively 5% and 10% is measured. This increase in $T_{\%}$ and an increase in d_G means that the correct goal is identified further from the position of the goal, but later in time with respect to the total length of the sequence. This can occur during data sharing when the human is tracked earlier after data sharing, but this extension of the sequence does not lead to an earlier estimation of the correct goal (for example because the human was walking towards another goal first). Then the total time from the first frame the human is tracked to a correct assessment is increased, while the absolute distance from the goal at which it is found can be further away than in the individual case.

Next, in configuration B and D a decrease in $T_{\%}$ is visible. In Configuration B an average decrease of 28% is denoted, while in Configuration D an average decrease of 11% is seen. On average the largest improvements are measured with configuration B, thus a breakdown of all sequences that belong to Configuration B is given in Table IX, separated based on the start positions of the humans. Each two consecutive rows in this table consist of the data from five different experiments. It is shown that the biggest improvements are made in Configuration g , which is where the humans start in the middle of the field. After examining these five experiments, it is clear the increase is attributed to three moments where a gap in the data is bridged by sharing data, two cases where the increased combined field of view enabled the detection of more humans reaching goals and one case where the addition of likelihoods resulted in a faster converging to the correct hypothesis. In the other configurations, these improvements are also present, but in Configuration a and f there are also three cases where a human is tracked from an earlier time because he is only in the field of view of one of the robots. In the next subsection the causes of these improvements are explained further, next to how these improvements impact the percentage of time and absolute distance to the goal at which the correct hypothesis first becomes dominant.

Table VII: The amount of correctly estimated sequences C_h clustered on starting positions of the soccer balls - **Goal: Balls**

Ball Config	Human Name	No Data Sharing		Data Sharing
		R1	R2	R1+R2
		C_h [-]	C_h [-]	C_h [-]
A	H1	22 of 31	15 of 31	22 of 31
A	H2	13 of 31	21 of 31	21 of 31
B	H1	16 of 20	8 of 20	18 of 20
B	H2	9 of 20	11 of 20	13 of 20
C	H1	13 of 27	12 of 27	21 of 27
C	H2	17 of 27	16 of 27	21 of 27
D	H1	12 of 20	5 of 20	16 of 20
D	H2	10 of 20	12 of 20	20 of 20

Table VIII: Percentagewise increases in $T_{\%}$ and d_G of the data sharing case with respect to the individual cases. These experiment results are clustered based on the starting positions of the soccer balls - **Goal: Balls**

Ball Config	Human Name	Differences of metrics			
		$T_{\%}$ [%]		d_G [-]	
		R1	R2	R1	R2
A	H1	-1.85	51.43	-5.66	14.77
A	H2	-19.64	-10.00	20.02	-1.77
B	H1	-28.57	-38.78	12.50	4.41
B	H2	-29.31	-10.87	87.48	21.91
C	H1	-3.12	16.98	13.20	-0.66
C	H2	35.00	-10.00	8.67	9.64
D	H1	5.36	-14.49	14.13	-11.47
D	H2	-26.32	13.51	4.54	46.65

Table IX: Percentagewise increases in $T_{\%}$ and d_G of the data sharing case with respect to the individual cases. These experiment results are clustered based on the starting positions of the humans in ball configuration B - **Goal: Balls**

Ball Config	Human Config	Human Name	Differences of metrics			
			$T_{\%}$ [%]		d_G [-]	
			R1	R2	R1	R2
B	a	H1	-6.33	-70.32	0.73	13.88
B	a	H2	-7.54	50.10	75.28	17.15
B	f	H1	-34.48	22.65	-7.62	19.89
B	f	H2	15.62	15.15	75.19	0.69
B	g	H1	-40.41	-46.69	36.03	18.26
B	g	H2	-70.94	-37.04	348.35	35.06
B	h	H1	-26.90	-23.23	21.16	-14.29
B	h	H2	1.00	-46.59	-0.38	29.28

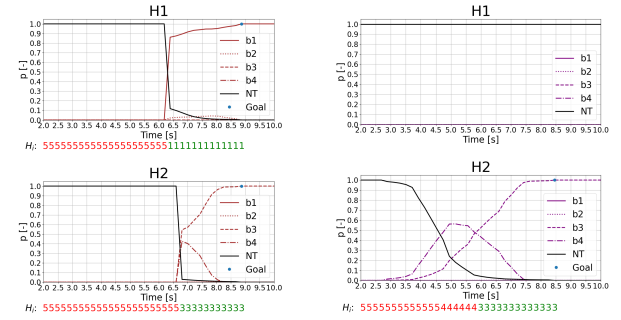
Improvements due to Data Sharing

In this subsection different situations are discussed where the sharing of data has a positive effect on the position sequences and accompanying probability distributions. These situations are 1) the extension of sequences due to a combined field of view, 2) the coverage of gaps in the sequence from different causes, 3) an increase in correctly estimated goals due to a combined field of view and 4) faster convergence due to the amplification of likelihoods in the overlapping part of the field of view of both robots. Each situation is illustrated with a clear example experiment.

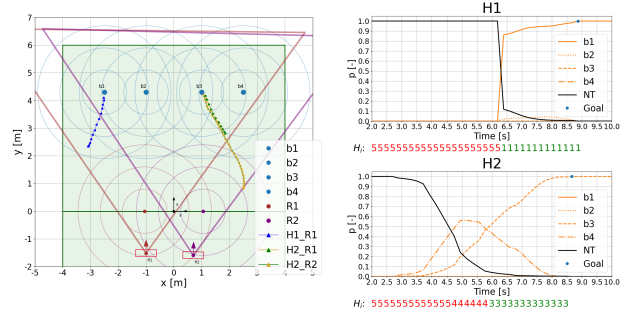
Sequence Extension due to Combined Field of View

The first improvement is the extension of the data range of the joined probability distribution. The field of view of each robot individually does not cover the full field and thus one robot can only track a human in a portion of the field. By adding the probability distributions together, a human can be tracked in a bigger area and thus the robots can potentially estimate the goal of the human earlier. An example of a situation where this is shown clearly is Experiment 31, of which the results of the Goal Estimation Pipeline are illustrated in Figure 17. Subfigure 17c shows that each human starts at the middle line, each at the other side of the robots. These positions are outside the fields of view of both robots. It is shown that the sequences extend back outside the field of view of robot $R1$ to the border of the field of view of robot $R2$. This is also visible in the probability distributions in Subfigure 17a and Subfigure 17b. For human $H2$ the first data point of robot $R1$ is registered around 6.8s. This is indicated by the No Data hypothesis H_{NT} dropping towards almost zero while the other hypotheses gain a visible positive value. For robot $R2$ there is already a faint rise in likelihood of hypothesis H_4 at 3.0s and it eventually switches to the correct hypothesis H_3 at around 5.8s. By combining the likelihoods (Subfigure 17d), human $H1$ is thus tracked about 3.8 second earlier with respect to robot $R1$ individually and estimates the correct goal 1.0s earlier than R_2 individually.

However, it should be noted that these extensions of sequences do not always result in a decrease in the percentage of total time and an increase in the absolute distance from the goal in the data sharing case. For example in Experiment 31, the percentage of total time is increasing from 10% for robot $R1$ to 54% for the joined case (see Appendix table D.14 for the data), although it is slightly improved when compared to the 56% of robot $R2$. This decrease happens because the sequence is extended, but in the added part first hypothesis H_3 is dominant. So the biggest part of the added sequence does not influence the time at which the correct hypothesis is found and thus the percentage of that time is increased with respect to the shorter sequence of robot $R1$ where the dominant hypothesis immediately is the correct one. The extension of sequences has little effect on d_G , because earlier tracking of the human does not influence the absolute distance from the goal where the correct hypothesis first becomes dominant.



(a) Individual likelihoods over time of both humans from R1 (b) Individual likelihoods over time of both humans from R2



(c) All position sequences from starting positions to the balls (d) Joined likelihoods over time for both humans

Figure 17: Example of how data sharing can lead to the extension of sequences. - **Experiment 31**

24 out of 98 experiments are identified where one or two sequences are extended by sharing data. They are grouped per goal configuration in Table X. It is apparent that the most sequence extensions happen in experiments with Configuration A and Configuration C. This is explained by the fact that most starting locations of the humans in these configurations are outside the field of view of the robots, while the goals are relatively in the middle of the field. This means the humans inadvertently have to come in the overlapping part of both views and thus the data sequence is extended based on the wider view towards the starting location of the human of one of the robots.

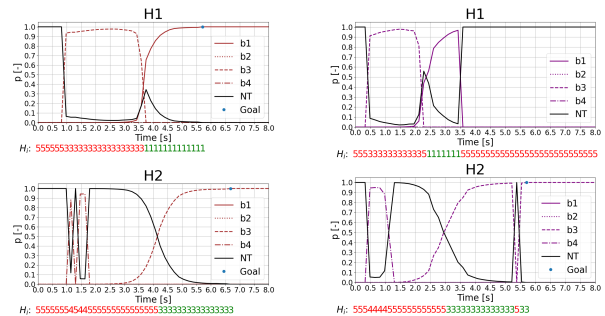
Furthermore, the most frequent presence of sequence extensions in Configuration A and Configuration C explains the average increase in $T_{\%}$ in those configurations in Table VIII, while the amount of correctly estimated goals C_g in Table VII is hardly influenced.

Table X: Experiment numbers per configuration of starting positions of the balls where sequence extensions are registered.

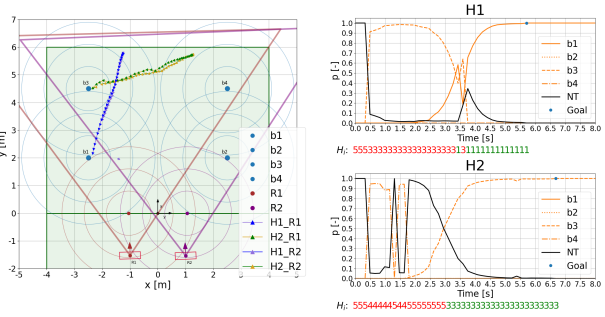
Configuration	Amount	Experiment Numbers
A	9	2, 28, 41, 59, 65, 67, 78, 90, 91
B	3	71, 74, 85
C	10	11, 18, 31, 37, 45, 75, 77, 79, 96, 98
D	2	60, 62

Sequence Data Gap Coverage

The second improvement is the bridging of gaps in the data sequences when two probability distributions are merged. An often small gap of only a few frames can appear in the data. This has a substantial influence on the percentage of total time at which the correct hypothesis is first dominant and absolute distance from the human towards the goal at that time frame. When a gap is present in the data, the No Data hypothesis H_{NT} is dominant during those frames. When after the data gap the sequence resumes, the first time at which the correct hypothesis becomes dominant is thus reset to the time frame after the data gap. By bridging this gap with the use of the data from the second robot, it is to be expected that large improvements are made to the involved metrics. Three causes for a data gap are treated: 1) a crossing of the sequences from the robots perspective, 2) a human leaves the field of view of one robot and later re-enters it and 3) unpredictable data loss due to inconsistencies in the camera, OpenPose or the rest of the pipeline.



(a) Individual likelihoods over time of both humans from R1 (b) Individual likelihoods over time of both humans from R2



(c) All position sequences from starting positions to the balls (d) Joined likelihoods over time for both humans

Figure 18: Example of how data sharing can lead to data gap coverage when sequences cross. - **Experiment 72**

In Subfigure 18c an example of a crossing in the position sequences from the perspective of the robots is shown. The resulting data gap in the individual likelihoods is shown in Subfigure 18b for human $H2$. When no position and orientation data is put into the intention prediction function,

the likelihoods for all hypotheses default to zero, while the likelihood for the No Data hypothesis remains constant. The No Data hypothesis is thus the only non-zero likelihood and after normalization it has a likelihood of 1.0. After the gap, the sequence continues and the goal is reached. In the joint probability distribution in Subfigure 18d, the data gap is bridged. The results is that the first time the correct hypothesis becomes dominant, is now at about 3.0s instead of at 5.5s for the case of the individual robot $R2$. This is further supported with data from Table D.16 in Appendix D-B, where robot $R2$ individually denotes a value for $T\%$ of 97% and 0.28m for d_G . In the data sharing case, $T\%$ decreases to 42% and d_G increases by 1048% to 2.94m. It is thus evident that the bridging of gaps due to crossings can massively impact the time and distance at which the correct hypothesis is first dominant, mostly because the data gaps due to crossings happen close to the goal and thus the gain in time and distance is highest.

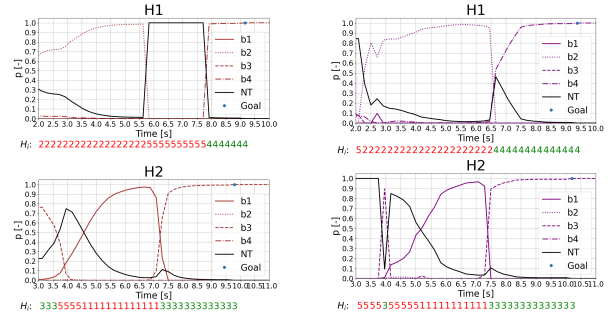
In Table XI all experiments where a gap in the data due to a crossing is bridged are shown. Situations with Configuration B and Configuration C contain most of these situations. This is explained by the fact that in these configurations most crossings with respect to the robot views happen in the overlapping part of the field of views of both robots. In Configuration B the humans can walk between the balls in the middle of the field and in Configuration C they have to walk from the middle line towards the goals next to each other on the far side of the field, while in Configuration A the middle is blocked by the goals itself and in Configuration D the goals are the most sparsely placed meaning that crossings do not often happen. Because the crossings happen in the overlapping part of the field of views, the data sequence is only shortly interrupted and can be more easily bridged with the help of the data from the other robot.

Table XI: Experiment numbers per configuration of starting positions of the balls where coverage of data gaps due to a crossing are registered.

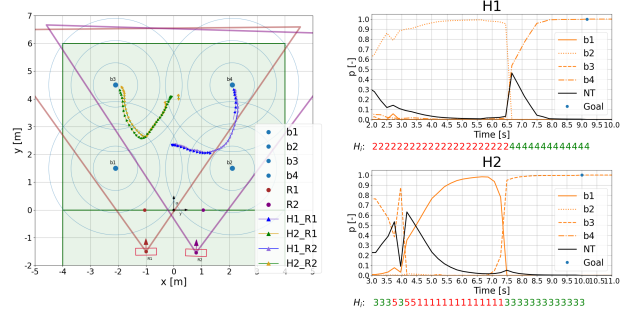
Configuration	Amount	Experiment Numbers
A	2	29, 32
B	4	13, 25, 29, 38
C	4	5, 27, 68, 75
D	2	62, 72

The next cause of a data gap occurring is when a human leaves the field of view of one robot and later re-enters the field of view on its path to a goal. A position sequence highlighting this is shown in Subfigure 19c. Human $H1$ leaves the field of view of robot $R1$ walking towards goal $b2$, after which he turns to the left and eventually reaches goal $b4$ again in the field of view of robot $R1$. The result is a gap in the probability distribution sequence of about 1.9s, which is visible in Subfigure 19a. Human $H1$ however never leaves the field of view of robot $R2$, so when the likelihoods are added

together in Subfigure 19d the data gap is entirely bridged by the data from robot $R2$. This lowers $T\%$ from about 7.8s in the individual case for robot $R1$ to about 7.4s in the joined case and the individual case of robot $R2$ (these parts are identical here because there is no data to be added from robot $R1$). The improvement is further supported by the decrease and increase respectively of $T\%$ and d_G from Table D.14 in Appendix D-B. Here a decrease from 83% to 67% is denoted for $T\%$, which is a decrease of 20% with respect to the individual case of robot $R1$. For d_G an increase from 0.99m to 2.20m is seen, which is a relative increase of 223%.



(a) Individual likelihoods over time of both humans from R1 (b) Individual likelihoods over time of both humans from R2



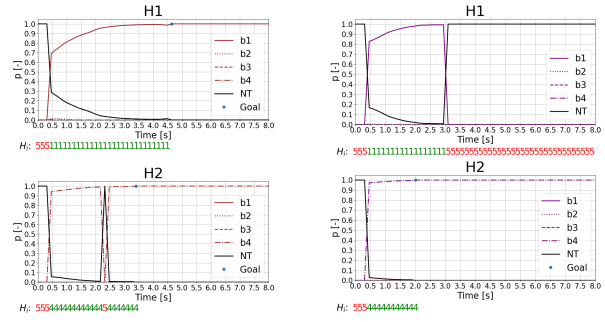
(c) All position sequences from starting positions to the balls (d) Joined likelihoods over time for both humans

Figure 19: Example of how data sharing can lead to data gap coverage when sequences span outside one robots field of view. - **Experiment 15**

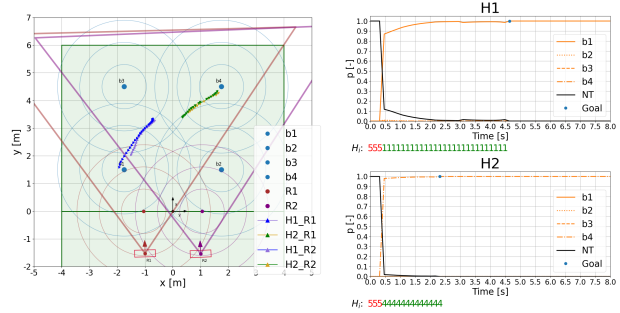
The distribution of occurring data gaps due to temporarily leaving the field of view of one robot in each configuration is shown in Table XII. The most occurrences are found with Configuration C. This is explained by the observation that both humans start at the middle line of the field, often outside the field of view of one or both robots. If the humans do not walk in a straight line towards their goal, they can easily leave the field of view of one robot when they pass through close to the position of the robot. However, the human then always needs to re-enter the field of view at some point, because all goals are inside the field of view of both robots in this configuration, thus resulting in the data gaps seen in these situations.

Table XII: Experiment numbers per configuration of starting positions of the balls where coverage of data gaps due to a human temporary leaving the field of view of one robot are registered.

Configuration	Amount	Experiment Numbers
A	0	-
B	1	15
C	3	11, 24, 27
D	2	30, 93



(a) Individual likelihoods over time of both humans from R1 (b) Individual likelihoods over time of both humans from R2



(c) All position sequences from starting positions to the balls (d) Joined likelihoods over time for both humans

Figure 20: Example of how data sharing can lead to data gap coverage when sequences are incomplete. - **Experiment 99**

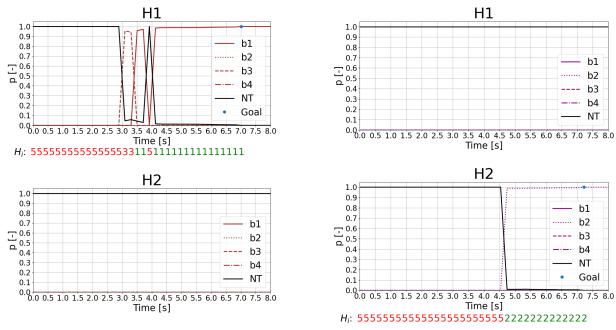
The last cause for the occurrence of a gap in the position sequence is unforeseen inconsistencies in the data gathering or processing pipelines, for example the absence of depth data at a specific spot in the files coming from the Kinect camera, no joints detected by OpenPose or wrongly adjusted joint positions due to the Far Joint Inferring algorithm (Subsection III-B2). One such an occurrence is shown in Figure 20. In Subfigure 20b it is shown that for human $H2$ a frame is missing at 2.3s. Due to this missing frame the time at which the correct hypothesis first becomes dominant moves from 0.4s to 2.4s. But due to the merging of the data of both robots, this data gap is visibly bridged in the graph of Subfigure 20d. In such, $T_{\%}$ is pushed back to the original 0.4s. This is again further supported by data from Table D.16 in Appendix D-B, where a decrease in $T_{\%}$ from 68% to 8% and an increase in d_G to goal from 0.69m to 1.59 is denoted. In the described cases, the sharing of data often has a big impact on the metrics, because the gaps can occur unexpectedly, and at any given time in the sequence. Mostly such a defect does not occur at the same time at the other robot, meaning that these gaps can be resolved in most situations by sharing data. In Table XIII the amount of unpredictable gaps in the data sequences per configuration is shown. In Configuration C the most inconsistencies occur, which is explained by the fact that the goals are all placed far from the camera where the rays used for the determination of depth are more sparse and thus more prone to inconsistencies.

Table XIII: Experiment numbers per configuration of starting positions of the balls where coverage of data gaps due to data loss are registered.

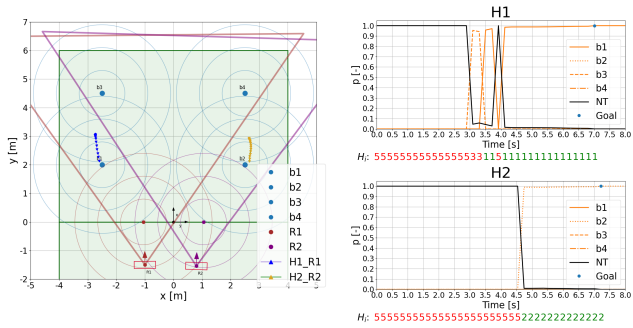
Configuration	Amount	Experiment Numbers
A	1	28
B	1	99
C	3	16, 37, 92
D	2	52, 95

Increase in Detected Goals due to Combined Field of View

The third improvement gained by the sharing of data is due to combining of two semi-overlapping fields of view. Each robot can observe the center and two farthest corners of the field with respect to the robots' positions, but closer to the middle line they can only observe their own side of the field. Due to the fact that the position of goal $b1$ in configurations B and D is outside or on the border of the field of view of robot $R2$, robot $R2$ can never mark that goal as reached based on the position of the human as a result of the lack of data in that area. However, by combining the data from the both robots the combined field of view is extended so all goals in the field are covered. The same holds for ball $b2$ and robot $R1$. An example situation is shown in Figure 21.



(a) Individual likelihoods over time of both humans from R1 (b) Individual likelihoods over time of both humans from R2



(c) All position sequences from starting positions to the balls (d) Joined likelihoods over time for both humans

Figure 21: Example of how data sharing can lead to registering more goals-reached. - **Experiment 1**

In Subfigure 21c it is shown that goal $b1$ and the starting position of human $H1$ lay both outside the field of view of robot $R2$. The same holds for $b2$, the starting position of human $H2$ and robot $R1$. This is also visible in Subfigure 21a and 21b, where there are no data points detected for the respective humans. When the likelihoods are combined in Subfigure 21d, both goals lay in the shared field of view and can thus be detected. This results in an immediate increase in the amount of correctly estimated goals. All experiments where such an increase is detected are shown in Table XIV. As expected, the most occurrences are detected in Configuration B and D, where the closest goals lay on or outside the field of view of one of the robots. Some occurrences are also detected in Configurations A and C. This is explained by the fact that the real life positions of the goals are not exactly the same in each experiment, and thus some cases exist where goals are placed too close to the field of view borders.

Table XIV: Experiment numbers per configuration of starting positions of the balls where more pick-ups of soccer balls are registered due to the combined field of view covering a bigger part of the field.

Configuration	Amount	Experiment Numbers
A	2	6, 78
B	5	36, 44, 71, 89, 99
C	2	18, 98
D	10	1, 17, 20, 39, 49, 52, 62, 72, 76, 93

Faster Convergence

The last improvement due to data sharing is the faster convergence towards the correct hypothesis by the amplification of the most dominant hypothesis in a situation where a human is in the field of view of both robots. If both robots are able to track the human, both supply a similar probability distribution to be added together during the data sharing part of the pipeline. When the likelihood of one hypothesis is big in comparison to the others, its magnitude is further amplified and the hypothesis may become dominant earlier than in both individual cases. An example is visible in Subfigure 22d, where the joined probability distribution of Experiment 51 is shown. In the graph for human $H2$, the correct hypothesis first becomes dominant at 1.2s, which is earlier than both the individual cases of robot $R1$ at about 1.7s and robot $R2$ at about 1.4s. All occurrences of noticeable faster convergence are shown in Table XV. Most cases are registered in Configuration A. This configuration can result in amplification of the joined likelihoods because the position of the goals are in the middle of the field where both robots can see the humans.

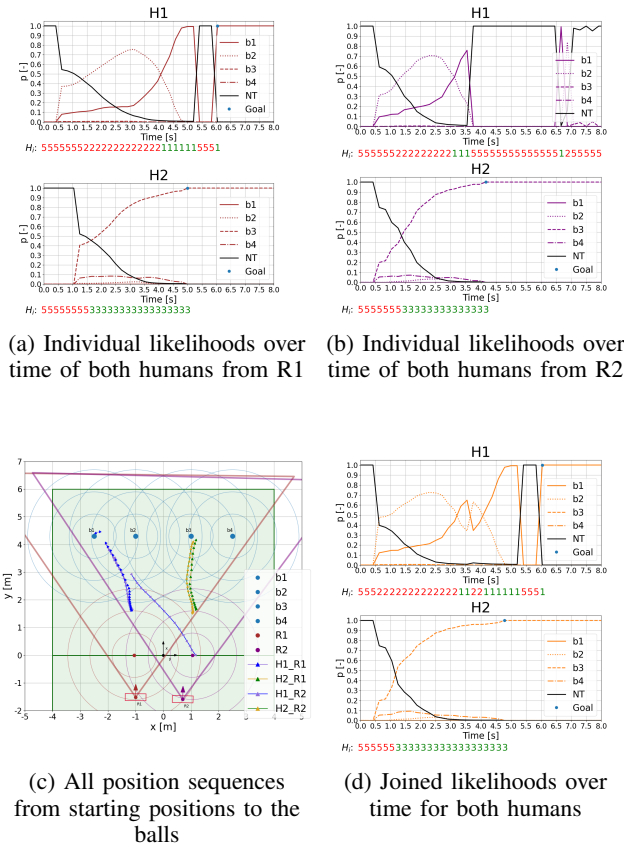


Figure 22: Example of how data sharing can lead to a faster convergence of the likelihood of the lead hypothesis. - **Experiment 51**

Table XV: Experiment numbers per configuration of starting positions of the balls where faster conversion of the likelihood of the leading hypothesis is registered.

Configuration	Amount	Experiment Numbers
A	3	2, 12, 22
B	0	-
C	1	51
D	1	30

VI. CONCLUSIONS

In this work a framework is provided in which different strategies for the intention prediction of multiple humans towards semantic goals can be tested for both an individual case and a case where data is shared between multiple robots. A pipeline is proposed for the estimation of shared probability distributions from a video feed of a RGBD-camera. First, a set of joint locations is estimated using the OpenPose library and these are then converted to real-world coordinates. Occluded joint positions are identified and repositioned with the proposed Far Joint Inferring algorithm. The position and orientation of each human is determined and a constant velocity Kalman filter is applied for trajectory smoothing. Next, the likelihood of each human going towards a goal is calculated with the multiplication of a 2D Gaussian function and a custom direction evaluation function. Lastly, the individual estimations of each robot are combined. To gather sufficient data to evaluate the proposed method, 98 experiments with each two persons, two robots and four semantic goals are carried out.

With the obtained data, four scenarios are identified where sharing data leads to a better assessment of the most likely semantic goal for the humans. These improvements are the earlier tracking of a human due to a combined field of view, the coverage of gaps in the data sequences, the increased detection of reached goals due to a combined field of view and the faster convergence towards the correct hypothesis by amplification of similar likelihoods.

It is thus concluded that even with relatively simple intention prediction and data sharing methods, already substantial improvements are observed when compared to the results of each robot individually. This may enable further research into the decentralization of autonomous robots in a warehousing environment.

VII. FUTURE WORK

Suggestions for future work as a result of this thesis include the following topics.

A. Exploration of Alternative Data Association Techniques

For the method used to associate new data points to the correct sequence, alternative methods can be explored to evaluate the likelihood of a new set of points belonging to one of multiple existing sets of points. Alternative methods can be easily compared within the proposed framework.

B. Exploration of Alternative Prediction Methods

In this thesis a 2D Gaussian function is used to estimate the likelihoods of all hypotheses in a scene. It is suggested to look into other prediction models such as the Hidden Markov model or an LSTM-network to estimate these likelihoods. The performance of each method can be compared to find the most suitable method for the application in different warehousing situations.

C. Exploration of Alternative Data Sharing Methods

In this thesis a data sharing method is used where the likelihoods of the same human estimated by multiple robots were added together. In future research there could also be looked at how the probability distributions of different humans estimated by the same robot influence each other in a multi-human multi-robot system.

D. Increased Amount of Tracked People

The proposed Goal Estimation Pipeline in this thesis is designed to work for two humans and two robots. However, there can be looked at expanding the this algorithm to incorporate more than two humans and more than two robots and the influence of this on the improvements in semantic goal estimation.

REFERENCES

- [1] D. B. Poudel, "Coordinating hundreds of cooperative, autonomous robots in a warehouse," *Jan*, vol. 27, no. 1-13, p. 26, 2013.
- [2] A. Prof. Dr. Lilienthal and M. Dr. Magnusson. (2021) Iliad project. [Online]. Available: <http://iliad-project.eu/>
- [3] Vanderlande. (2022) Vanderlande pallet av. [Online]. Available: <https://www.vanderlande.com/systems/internal-transport/pallet-av/>
- [4] A. Maoudj and A. L. Christensen, "Decentralized multi-agent path finding in warehouse environments for fleets of mobile robots with limited communication range," in *International Conference on Swarm Intelligence*. Springer, 2022, pp. 104–116.
- [5] A. Sayed, S. Scales, A. Fox, J. Stone, and S. Crews, "Reinforcement learning on modular robots using real-time decentralized deconfliction in a warehouse environment," in *ASME International Mechanical Engineering Congress and Exposition*, vol. 85604. American Society of Mechanical Engineers, 2021, p. V006T06A036.
- [6] M. S. L. Technical Committee, "Middle size robot league – rules and regulations," *Middle Size Robot League - Rules and Regulations for 2022*, pp. COMPETITION RULE 7 – Human player, 2022.
- [7] H. Razali, T. Mordan, and A. Alahi, "Pedestrian intention prediction: A convolutional bottom-up multi-task approach," *Transportation Research Part C: Emerging Technologies*, vol. 130, p. 103259, 2021. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0968090X21002710>
- [8] D. Wei, L. Chen, L. Zhao, H. Zhou, and B. Huang, "A vision-based measure of environmental effects on inferring human intention during human robot interaction," *IEEE Sensors Journal*, 2021.
- [9] R. Song, "Driver intention prediction using model-added bayesian network," *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, vol. 235, no. 5, pp. 1236–1244, 2021. [Online]. Available: <https://doi.org/10.1177/0954407020968967>
- [10] Z. Pei, X. Qi, Y. Zhang, M. Ma, and Y.-H. Yang, "Human trajectory prediction in crowded scene using social-affinity long short-term memory," *Pattern Recognition*, vol. 93, pp. 273–282, 2019.
- [11] L. Russo, M. Terlizzi, M. Tipaldi, and L. Glielmo, "A reinforcement learning approach for pedestrian collision avoidance and trajectory tracking in autonomous driving systems," in *2021 5th International Conference on Control and Fault-Tolerant Systems (SysTol)*, 2021, pp. 44–49.
- [12] R. Esmeli, M. Bader-El-Den, and H. Abdullahi, "Towards early purchase intention prediction in online session based retailing systems," *Electronic Markets*, vol. 31, no. 3, pp. 697–715, 2021.
- [13] S. Liu, K. Zheng, L. Zhao, and P. Fan, "A driving intention prediction method based on hidden markov model for autonomous driving," *Computer Communications*, vol. 157, pp. 143–149, 2020. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0140366419316639>
- [14] T. Petković, D. Puljiz, I. Marković, and B. Hein, "Human intention estimation based on hidden markov model motion validation for safe flexible robotized warehouses," *Robotics and Computer-Integrated Manufacturing*, vol. 57, pp. 182–196, 2019.
- [15] G. Bijlenga, M. v. d. Molengraft, and W. Houtman, "Human maneuvering intention estimation," Master's thesis, Eindhoven University of Technology, 2021.
- [16] G. Boesch. (2022) A guide to openpose in 2022. [Online]. Available: <https://viso.ai/deep-learning/openpose/>
- [17] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 7291–7299.
- [18] C. Jekel. (2018) Least squares sphere fit. [Online]. Available: <https://jekel.me/2015/Least-Squares-Sphere-Fit/>

Appendices

APPENDIX A SUPPORT METHODS

In this appendix two methods are explained that are used to support accuracy of the Goal Estimation Pipeline. In Appendix A-A a basic method is explained for retrieving the $[x, y, z]$ -positions of the soccer balls. After that, in Appendix A-B it is explained how these acquired positions of the balls are used to estimate the rotation- and position-offset of the Kinects during the experiments. It is also explained how the tilt of the Kinect is calculated by fitting a line through a set of points in the floor. By estimating the position offset, rotation offset and tilt offset of the robots, it is made sure both robots operate in the same world coordinate system.

A. Ball Recognition

During the experiments, the semantic goals are represented by four soccer balls. All soccer balls are a yellow color, so they can be found back in the pointcloud by analyzing the color frame of the Kinects. The complete algorithm is displayed in Algorithm 6. The inputs for the algorithm are the pointcloud of the Kinect, the color frame from the Kinect and the $[x, y, z]$ -positions of all four balls. First, the 250 points with a RGB-value closest to yellow are retrieved from the color matrix of the Kinect and the accompanying $[x, y, z]$ -values are located in the pointcloud (row 1). Then, these points are split into four sets based on the closest proximity to one of the goals (row 2). Next, because all points of a set lay on the exterior of the sphere, a sphere can be fit through the points using a Linear Least Squares approximation of

$$\sum_{k=1}^n ((v_k - m)^2 - r^2)^2, \quad (15)$$

where v_k are the points in the set belonging to a ball, m is the center of the sphere and r is the radius (row 4) [18]. Lastly, the center and radius for each ball is returned (row 5). These estimated ball positions are not used for the prediction of human intention (because the goal location are assumed to be known), but for calculating the position- and rotation-offset of each Kinect during the experiments. This is further elaborated on in the next subsection.

Algorithm 6 Ball recognition

Require: pointcloud: all $[x, y, z]$ points in world coordinates

Require: color frame: RGB values for each point

Require: ball positions: $[x, y, z]$ positions of the balls in world coordinates

- 1: Take $[x, y, z]$ values of 250 points with a color closest to yellow
 - 2: Divide points in four sets based on proximity to one of the balls
 - 3: **for** each ball **do**
 - 4: Fit sphere through the points of the accompanying set using Linear Least Squares
 - 5: **end for**
 - 6: **return** center $[x, y, z]$ in world coordinates
 - 7: **return** radius $[m]$
-

B. Syncing of Coordinate Systems

In order for the data of the two robots to be comparable, the two robots have to operate in the same coordinate system. If this is not the case, the estimations for joint positions in Algorithm 1 would too different although the two robots are witnessing the same actions. During the experiments the Kinect camera's of both robots were carefully placed at their designated spots, but small errors in position and orientation are always possible. In order to determine these errors, the results of the ball recognition algorithm (Algorithm 6) and the next algorithm are used to roughly estimate the values for the offsets, after which they are tuned by hand until a sufficient similarity in coordinate systems is achieved.

In Algorithm 7 it is explained how the tilt of each Kinect is estimated. First, the 1000 closest points to the vector extending forward from the Kinect are selected (row 1). A 1D polynomial is then fitted through these points using a Linear Least Squares approximation (row 2). This outputs a vertical offset and a slope. The angle between the vector of the slope and the vector extending forward from the Kinect is then calculated (row 3). These values are used for the y -offset and the tilt ϕ respectively.

Algorithm 7 Tilt Estimation

Require: pointcloud: all $[x, y, z]$ points in world coordinates

- 1: Select 1000 closest points to the vector extending forward from the Kinect
 - 2: Fit 1D polynomial through y -coordinates of points in set
 - 3: Calculate angle ϕ of fitted vector with vector extending forward from the Kinect
 - 4: **return** tilt ϕ , y -offset
-

The estimated offsets are displayed in Table A.1. Because the experiment setup was build anew every day when experiments were held, the offset values also differ between days. By estimation of these offsets, the position sequences estimated by both robots can be compared in a more accurate manner.

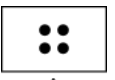
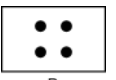
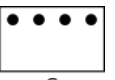

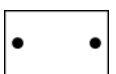
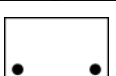
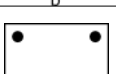
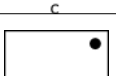
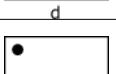
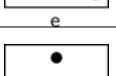
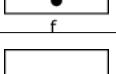
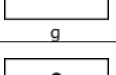
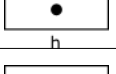
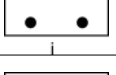
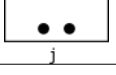
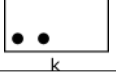
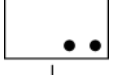
Experiment day	robot	x -offset [m]	y -offset [m]	z -offset [m]	ϕ -offset [°]	ψ -offset [°]	θ -offset [°]
Day 1	R1	0.0	-0.03	0.0	-1.63	0.0	-0.55
	R2	0.0	0.04	0.0	2.69	0.0	1.71
Day 2	R1	0.0	-0.04	0.0	-1.55	0.0	0.66
	R2	0.0	-0.01	0.0	2.17	0.0	1.46
Day 3	R1	0.0	-0.04	0.0	-1.15	0.0	-1.45
	R2	0.0	-0.04	0.0	-0.68	0.0	-2.35

Table A.1: Estimated offset of each Kinect during each day experiments were conducted. The x -, y - and z -offset are in world coordinates, ϕ is the rotation around the x -axis (or tilt), ψ is the rotation around the y -axis (or roll) and θ is the rotation around the z -axis (or rotation), all in world coordinates.

APPENDIX B
EXPERIMENT CONFIGURATIONS

In Table B.1 it is shown which combinations of configurations for the starting positions of the soccer balls and starting positions for the humans are used during the experiments. Which combination is used in each experiment is displayed in Table D.13.

Table B.1: Contingency matrix of ball configurations A-D and human configurations a - m .

Ball configs \Rightarrow Human configs \Downarrow				
	A	B	C	D
 a	✓	✓	✗	✓
 b	✓	✗	✓	✗
 c	✓	✗	✗	✗
 d	✓	✗	✗	✗
 e	✓	✗	✗	✗
 f	✓	✓	✗	✗
 g	✗	✓	✗	✓
 h	✗	✓	✗	✗
 i	✗	✗	✓	✗
 j	✗	✗	✓	✓
 k	✗	✗	✓	✗
 l	✗	✗	✓	✗
 m	✗	✗	✗	✓

APPENDIX C
REPOSITORY STRUCTURE

The repository of this thesis is open source and can be found at https://gitlab.tue.nl/et_projects/svintentionrecognition. The following components are shared and available:

- All scripts used to perform the experiments, including a instruction manual for installation of all Ubuntu packages and the drivers needed for the Kinects.
- Figures of the used setups for each experiment.
- The video data obtained from the experiments, structured per experiment and pre-analyzed by OpenPose.
- A complete library of the Goal Estimation Pipeline. All classes and function are fully commented and section interesting for future work are highlighted.
- 3D and 2D figures of the intention prediction, the position and orientation estimation and probability distributions.

APPENDIX D
RESULT TABLES

In this appendix all tables with results of the experiments are displayed.

A. Clustered Tables

In this subsection all tables are displayed based on different relevant clustering of the raw data obtained from the experiments. In all tables the following symbols are present: C_h here can be True or False based on if the goal of the sequence is estimated correctly, $T_{\%}$ is the percentage of the total time the human is tracked at which the correct hypothesis first becomes dominant and d_G is the absolute distance to the goal when the correct hypothesis first becomes dominant.

Overall Data

In Table D.1 the summed values for C_h are displayed. In Table D.2 the mean of the values for $T_{\%}$ and d_G obtained from the experiments is displayed. Both are clustered over all experiments. This is done to assess the overall performance of the sharing of data on the metrics.

Table D.1: The mean of the amount of correctly assessed sequences C_h in all experiments - **Goal: Balls**

Config	No Data Sharing		Data Sharing
	R1	R2	R1+R2
	C_h [-]	C_h [-]	C_h [-]
all	112 of 196	100 of 196	152 of 196

Table D.2: The mean over all experiments of the percentage of total time $T_{\%}$ and absolute distance d_G at which a correct assessment is made - **Goal: Balls**

Config	No Data Sharing				Data Sharing	
	R1		R2		R1+R2	
	$T_{\%}$ [%]	d_G [m]	$T_{\%}$ [%]	d_G [m]	$T_{\%}$ [%]	d_G [m]
all	52	1.60	49	1.64	49	1.81

Data Clustered on Humans

In Table D.3 the summed values for C_h are displayed. In Table D.4 the mean of the values for $T_{\%}$ and d_G obtained from the experiments is displayed. Both are clustered based on the two humans present in the experiments. This is done to determine if irregularities are existent belonging to one specific human and if the sharing of data favours one specific human. Both statements are not true.

Table D.3: The amount of correctly assessed sequences C_h , clustered based on the two humans present in the experiments - **Goal: Balls**

Human Name	No Data Sharing		Data Sharing
	R1	R2	R1+R2
	C_h [-]	C_h [-]	C_h [-]
H1	63 of 98	40 of 98	77 of 98
H2	49 of 98	60 of 98	75 of 98

Table D.4: The mean of the experiment results for $T_{\%}$ and d_G , clustered based on the two humans present in the experiments - **Goal: Balls**

Human Name	No Data Sharing				Data Sharing	
	R1		R2		R1+R2	
	$T_{\%}$ [%]	d_G [m]	$T_{\%}$ [%]	d_G [m]	$T_{\%}$ [%]	d_G [m]
H1	53	1.65	48	1.69	51	1.76
H2	51	1.54	50	1.60	46	1.85

Data Clustered on Crossing / No-crossing

In Table D.5 the summed values for C_h are displayed. In Table D.6 the mean of the values for $T_{\%}$ and d_G obtained from the experiments is displayed. Both are clustered based on if a crossing of trajectories happened in that particular experiment. This is done to evaluate if the sharing of data results in more improvement in situations where crossings are present. From the data it is derived that this is indeed the case.

Table D.5: Experiment results for C_h , clustered based on if a crossing of trajectories happened in that particular experiment - **Goal: Balls**

Crossing in experiment	No Data Sharing		Data Sharing
	R1	R2	R1+R2
	C_h [-]	C_h [-]	C_h [-]
True	57 of 100	53 of 100	75 of 100
False	55 of 96	47 of 96	77 of 96

Table D.6: The mean of the experiment results for $T_{\%}$ and d_G , clustered based on if a crossing of trajectories happened in that particular experiment - **Goal: Balls**

Crossing in experiment	No Data Sharing				Data Sharing	
	R1		R2		R1+R2	
	$T_{\%}$ [%]	d_G [m]	$T_{\%}$ [%]	d_G [m]	$T_{\%}$ [%]	d_G [m]
True	61	1.57	61	1.59	56	1.88
False	43	1.63	35	1.69	42	1.74

Data Clustered on Different Ball Starting Positions

In Table D.7 the summed values for C_h are displayed. In Table D.8 the mean of the values for $T_{\%}$ and d_G obtained from the experiments is displayed. Both are clustered on starting positions of the soccer balls. This is done to evaluate if the sharing of data leads to more improvement in particular configurations of goals. It is concluded that the most improvement by sharing data is gained in Configuration B.

Table D.7: Experiment results for C_h , clustered on starting positions of the soccer balls - **Goal: Balls**

Ball Config	Human Name	No Data Sharing		Data Sharing
		R1	R2	R1+R2
		C_h [-]	C_h [-]	C_h [-]
A	H1	22 of 31	15 of 31	22 of 31
A	H2	13 of 31	21 of 31	21 of 31
B	H1	16 of 20	8 of 20	18 of 20
B	H2	9 of 20	11 of 20	13 of 20
C	H1	13 of 27	12 of 27	21 of 27
C	H2	17 of 27	16 of 27	21 of 27
D	H1	12 of 20	5 of 20	16 of 20
D	H2	10 of 20	12 of 20	20 of 20

Table D.8: The mean of the experiment results for $T_{\%}$ and d_G , clustered on starting positions of the soccer balls - **Goal: Balls**

Ball Config	Human Name	No Data Sharing				Data Sharing	
		R1		R2		R1+R2	
		$T_{\%}$ [%]	d_G [m]	$T_{\%}$ [%]	d_G [m]	$T_{\%}$ [%]	d_G [m]
A	H1	54	1.71	35	1.40	53	1.61
A	H2	56	1.37	50	1.67	45	1.65
B	H1	42	1.67	49	1.80	30	1.88
B	H2	58	1.01	46	1.55	41	1.89
C	H1	64	1.63	53	1.86	62	1.85
C	H2	40	1.72	60	1.70	54	1.87
D	H1	56	1.52	69	1.96	59	1.73
D	H2	57	1.95	37	1.39	42	2.04

Data Clustered on Different Human Starting Positions

In Table D.9 the summed values for C_h are displayed. In Table D.10 the mean of the values for $T_\%$ and d_G obtained from the experiments is displayed. Both are clustered based on starting positions of the humans. This is done to evaluate if the starting positions of the humans have a decisive influence on the improvements due to data sharing. Although most rows show an improvement in the data sharing case, no clear conclusions are derived from this clustering of data.

Table D.9: Experiment results for C_h , clustered based on starting positions of the humans - **Goal: Balls**

Human Config	Human Name	No Data Sharing		Data Sharing
		R1	R2	R1+R2
		C_h [-]	C_h [-]	C_h [-]
a	H1	12 of 14	5 of 14	12 of 14
a	H2	5 of 14	7 of 14	10 of 14
b	H1	5 of 12	2 of 12	7 of 12
b	H2	6 of 12	5 of 12	9 of 12
c	H1	2 of 5	2 of 5	3 of 5
c	H2	4 of 5	4 of 5	4 of 5
d	H1	5 of 6	4 of 6	6 of 6
d	H2	2 of 6	5 of 6	4 of 6
e	H1	4 of 5	2 of 5	3 of 5
e	H2	1 of 5	3 of 5	3 of 5
f	H1	8 of 10	5 of 10	9 of 10
f	H2	6 of 10	7 of 10	8 of 10
g	H1	8 of 10	4 of 10	9 of 10
g	H2	5 of 10	8 of 10	9 of 10
h	H1	3 of 5	2 of 5	3 of 5
h	H2	2 of 5	2 of 5	2 of 5
i	H1	3 of 6	3 of 6	5 of 6
i	H2	4 of 6	4 of 6	4 of 6
j	H1	5 of 10	4 of 10	7 of 10
j	H2	5 of 10	8 of 10	10 of 10
k	H1	2 of 5	2 of 5	3 of 5
k	H2	1 of 5	2 of 5	2 of 5
l	H1	2 of 5	4 of 5	5 of 5
l	H2	3 of 5	3 of 5	5 of 5
m	H1	4 of 5	1 of 5	5 of 5
m	H2	5 of 5	2 of 5	5 of 5

Table D.10: The mean of the experiment results for $T_\%$ and d_G , clustered based on starting positions of the humans - **Goal: Balls**

Human Config	Human Name	No Data Sharing				Data Sharing	
		R1		R2		R1+R2	
		$T_\%$ [%]	d_G [m]	$T_\%$ [%]	d_G [m]	$T_\%$ [%]	d_G [m]
a	H1	43	1.64	56	1.70	38	1.66
a	H2	57	1.56	26	1.85	40	2.03
b	H1	50	1.88	39	1.77	49	1.79
b	H2	52	1.34	59	1.33	52	1.42
c	H1	42	2.44	27	1.72	58	1.88
c	H2	44	1.38	47	1.90	51	1.86
d	H1	37	1.62	29	1.29	48	1.45
d	H2	49	1.43	64	1.48	59	1.45
e	H1	78	1.35	54	0.50	83	1.28
e	H2	57	1.46	46	1.53	35	1.60
f	H1	48	1.78	29	1.57	32	1.82
f	H2	62	1.27	50	1.70	51	1.72
g	H1	37	1.82	60	1.73	29	2.10
g	H2	84	0.77	34	1.58	33	2.17
h	H1	56	1.62	53	2.30	41	1.97
h	H2	38	1.59	72	1.22	39	1.58
i	H1	74	1.18	36	2.38	70	1.57
i	H2	28	1.96	56	1.87	37	2.35
j	H1	85	1.10	46	2.20	71	1.93
j	H2	26	2.31	50	1.86	47	2.02
k	H1	53	2.39	43	1.34	57	2.17
k	H2	75	0.90	60	1.40	56	1.85
l	H1	61	1.89	81	1.47	74	1.85
l	H2	43	1.68	73	1.37	68	1.34
m	H1	75	1.24	69	2.08	75	1.39
m	H2	52	1.98	58	0.40	41	2.31

Data Clustered on Different Experiment Setups

In Table D.11 the summed values for C_h are displayed. In Table D.12 the mean of the values for $T_{\%}$ and d_G obtained from the experiments is displayed. Both are clustered based on the combination of the starting positions of the soccer balls and humans. This is done to evaluate if a particular combination of starting positions for humans and soccer balls shows a decisive bigger improvement of the metrics than other combinations. Furthermore, parts of these tables in combination with the accompanying probability distributions are used to identify why Configuration B holds the largest improvement when data is shared.

Table D.11: Experiment results for C_h , clustered based on the combination of the starting positions of the soccer balls and humans - **Goal: Balls**

Ball Config	Human Config	Human Name	No Data Sharing		Data Sharing
			R1	R2	R1+R2
			C_h [-]	C_h [-]	C_h [-]
A	a	H1	4 of 4	3 of 4	4 of 4
A	a	H2	1 of 4	3 of 4	2 of 4
A	b	H1	2 of 6	1 of 6	2 of 6
A	b	H2	1 of 6	2 of 6	4 of 6
A	c	H1	2 of 5	2 of 5	3 of 5
A	c	H2	4 of 5	4 of 5	4 of 5
A	d	H1	5 of 6	4 of 6	6 of 6
A	d	H2	2 of 6	5 of 6	4 of 6
A	e	H1	4 of 5	2 of 5	3 of 5
A	e	H2	1 of 5	3 of 5	3 of 5
A	f	H1	5 of 5	3 of 5	4 of 5
A	f	H2	4 of 5	4 of 5	4 of 5
B	a	H1	5 of 5	1 of 5	5 of 5
B	a	H2	2 of 5	2 of 5	3 of 5
B	f	H1	3 of 5	2 of 5	5 of 5
B	f	H2	2 of 5	3 of 5	4 of 5
B	g	H1	5 of 5	3 of 5	5 of 5
B	g	H2	3 of 5	4 of 5	4 of 5
B	h	H1	3 of 5	2 of 5	3 of 5
B	h	H2	2 of 5	2 of 5	2 of 5
C	b	H1	3 of 6	1 of 6	5 of 6
C	b	H2	5 of 6	3 of 6	5 of 6
C	i	H1	3 of 6	3 of 6	5 of 6
C	i	H2	4 of 6	4 of 6	4 of 6
C	j	H1	3 of 5	2 of 5	3 of 5
C	j	H2	4 of 5	4 of 5	5 of 5
C	k	H1	2 of 5	2 of 5	3 of 5
C	k	H2	1 of 5	2 of 5	2 of 5
C	l	H1	2 of 5	4 of 5	5 of 5
C	l	H2	3 of 5	3 of 5	5 of 5
D	a	H1	3 of 5	1 of 5	3 of 5
D	a	H2	2 of 5	2 of 5	5 of 5
D	g	H1	3 of 5	1 of 5	4 of 5
D	g	H2	2 of 5	4 of 5	5 of 5
D	j	H1	2 of 5	2 of 5	4 of 5
D	j	H2	1 of 5	4 of 5	5 of 5
D	m	H1	4 of 5	1 of 5	5 of 5
D	m	H2	5 of 5	2 of 5	5 of 5

Table D.12: The mean of the experiment results for $T_{\%}$ and d_G , clustered based on the combination of the starting positions of the soccer balls and humans - **Goal: Balls**

Ball Config	Human Config	Human Name	No Data Sharing				Data Sharing	
			R1		R2		R1+R2	
			$T_{\%}$ [%]	d_G [m]	$T_{\%}$ [%]	d_G [m]	$T_{\%}$ [%]	d_G [m]
A	a	H1	64	1.94	50	1.55	54	1.78
A	a	H2	83	0.44	40	1.90	46	2.02
A	b	H1	62	1.62	7	2.04	57	1.54
A	b	H2	44	1.67	49	1.32	34	1.24
A	c	H1	42	2.44	27	1.72	58	1.88
A	c	H2	44	1.38	47	1.90	51	1.86
A	d	H1	37	1.62	29	1.29	48	1.45
A	d	H2	49	1.43	64	1.48	59	1.45
A	e	H1	78	1.35	54	0.50	83	1.28
A	e	H2	57	1.46	46	1.53	35	1.60
A	f	H1	45	1.64	29	1.58	30	1.76
A	f	H2	67	1.46	49	1.82	43	1.88
B	a	H1	20	1.64	64	1.45	19	1.65
B	a	H2	42	1.41	26	2.11	39	2.47
B	f	H1	53	2.02	28	1.56	34	1.87
B	f	H2	51	0.89	51	1.54	59	1.55
B	g	H1	50	1.52	55	1.75	30	2.07
B	g	H2	86	0.43	40	1.44	25	1.94
B	h	H1	56	1.62	53	2.30	41	1.97
B	h	H2	38	1.59	72	1.22	39	1.58
C	b	H1	42	2.05	71	1.49	46	1.89
C	b	H2	53	1.28	66	1.34	63	1.57
C	i	H1	74	1.18	36	2.38	70	1.57
C	i	H2	28	1.96	56	1.87	37	2.35
C	j	H1	86	0.99	23	2.57	63	1.90
C	j	H2	25	2.27	51	2.21	45	2.31
C	k	H1	53	2.39	43	1.34	57	2.17
C	k	H2	75	0.90	60	1.40	56	1.85
C	l	H1	61	1.89	81	1.47	74	1.85
C	l	H2	43	1.68	73	1.37	68	1.34
D	a	H1	53	1.25	65	2.38	48	1.49
D	a	H2	60	2.28	7	1.50	39	1.77
D	g	H1	15	2.33	72	1.68	29	2.14
D	g	H2	79	1.27	29	1.71	39	2.34
D	j	H1	82	1.28	69	1.83	77	1.94
D	j	H2	30	2.49	49	1.51	48	1.73
D	m	H1	75	1.24	69	2.08	75	1.39
D	m	H2	52	1.98	58	0.40	41	2.31

Experiment Numbers per Situation

In Table D.13 all experiments corresponding to each combination of human starting positions and ball starting positions are shown. The amount of experiment numbers in a row thus indicates how many times the same situations was tested in an experiment. However, the trajectories obtained as a result are not exactly the same for each repeated experiment. The experiments are split into a column where a crossing happened during the experiments and a group where it did not happen. This is done because this feature could not be automated and thus had to be checked by hand.

Table D.13: Experiment numbers corresponding to each experiment situation - **Goal: Balls**

Ball Config	Human Config	Corresponding Experiments	
		Crossing	No Crossing
A	a	19, 40, 90	6
A	b	14, 26, 56, 65	80, 101
A	c	-	4, 35, 50, 67, 91
A	d	9, 22	43, 59, 78, 97
A	e	2, 28	41, 63, 86
A	f	12, 32, 54, 61, 88	-
B	a	38	10, 44, 71, 85
B	f	36, 48, 74, 83	8
B	g	13, 25, 66	89, 99
B	h	29, 46, 57, 87	15
C	b	3, 73, 82	31, 58, 98
C	i	5, 27	45, 64, 79, 94
C	j	51, 96	18, 37, 77
C	k	100	11, 24, 68, 84
C	l	33, 42, 75, 92	16
D	a	62, 81	1, 23, 52
D	g	21, 49, 60	17, 76
D	j	30, 69, 93	20, 39
D	m	72, 95	7, 34, 47

B. Raw Data Tables

In this appendix the unclustered results of all experiments are displayed. The data is split in the individual cases and the cases where data is shared. C_h here can be True or False based on if the goal of the sequence is estimated correctly, $T\%$ is the percentage of the total time the human is tracked at which the correct hypothesis first becomes dominant and d_G is the absolute distance to the goal when the correct hypothesis first becomes dominant.

Table D.14: Raw data table (part 1) - **Goal: Balls**

Ball Config	Human Config	Experiment Number	Human Name	Crossing	No Data Sharing						Data Sharing		
					R1			R2			R1+R2		
					C_h [-]	$T\%$ [%]	d_G [m]	C_h [-]	$T\%$ [%]	d_G [m]	C_h [-]	$T\%$ [%]	d_G [m]
D	a	1	H1	False	True	26.32	1.09	False	-	-	True	26.32	1.09
D	a	1	H2	False	False	-	-	True	8.33	0.94	True	8.33	0.94
A	e	2	H1	True	True	72.73	1.69	False	-	-	True	86.96	0.91
A	e	2	H2	True	True	57.14	1.46	True	77.78	1.04	True	80.77	1.25
C	b	3	H1	True	True	47.37	2.27	False	-	-	True	42.86	2.27
C	b	3	H2	True	True	78.26	1.35	False	-	-	True	91.67	0.92
A	c	4	H1	False	True	42.86	2.58	True	7.69	2.17	True	42.86	2.58
A	c	4	H2	False	True	22.22	1.68	True	69.23	1.86	True	73.08	1.68
C	i	5	H1	True	True	77.78	0.81	True	29.17	2.45	True	48.15	2.41
C	i	5	H2	True	True	35.71	1.99	True	64.52	2.16	True	64.52	2.07
A	a	6	H1	False	True	54.17	2.08	True	25.00	2.06	True	13.04	2.01
A	a	6	H2	False	False	-	-	True	13.64	2.24	True	13.64	2.24
D	m	7	H1	False	True	59.09	2.24	False	-	-	True	59.09	2.05
D	m	7	H2	False	True	51.72	2.45	False	-	-	True	41.38	2.75
B	f	8	H1	False	False	-	-	True	12.50	0.79	True	1.05	0.82
B	f	8	H2	False	True	14.29	0.72	True	20.00	0.62	True	14.29	0.67
A	d	9	H1	True	True	50.00	1.64	True	20.00	1.63	True	41.67	1.70
A	d	9	H2	True	False	-	-	False	-	-	False	-	-
B	a	10	H1	False	True	8.33	1.11	False	-	-	True	8.33	1.14
B	a	10	H2	False	False	-	-	True	45.45	2.23	True	45.45	2.57
C	k	11	H1	False	True	52.94	2.54	True	20.00	1.20	True	52.94	2.54
C	k	11	H2	False	True	75.00	0.90	True	82.14	0.47	True	75.00	0.87
A	f	12	H1	True	True	20.00	1.36	True	7.14	1.33	True	6.67	1.35
A	f	12	H2	True	True	62.86	2.12	True	59.38	2.13	True	61.76	2.03
B	g	13	H1	True	True	41.94	2.57	True	74.07	1.23	True	26.67	2.69
B	g	13	H2	True	True	95.83	0.34	True	100.00	0.23	True	36.96	2.14
A	b	14	H1	True	False	-	-	False	-	-	False	-	-
A	b	14	H2	True	False	-	-	False	-	-	False	-	-
B	h	15	H1	False	True	83.33	0.99	True	63.89	2.20	True	66.67	2.20
B	h	15	H2	False	True	69.23	1.95	True	67.50	1.90	True	70.00	1.87
C	l	16	H1	False	True	74.29	1.65	False	-	-	True	74.29	1.59
C	l	16	H2	False	True	11.11	1.51	True	52.17	1.97	True	52.17	1.97
D	g	17	H1	False	True	5.26	2.02	False	-	-	True	5.00	1.84
D	g	17	H2	False	True	58.06	2.46	False	-	-	True	53.12	2.64
C	j	18	H1	False	True	61.54	2.52	True	26.09	2.53	True	50.00	2.84
C	j	18	H2	False	False	-	-	True	62.50	2.32	True	62.50	2.32
A	a	19	H1	True	True	70.00	1.94	True	43.75	1.83	True	70.00	1.80
A	a	19	H2	True	True	82.76	0.44	True	92.50	0.87	True	78.57	1.81
D	j	20	H1	False	True	65.00	2.34	False	-	-	True	65.00	2.34
D	j	20	H2	False	False	-	-	True	29.17	1.80	True	29.17	1.80
D	g	21	H1	True	False	-	-	True	72.00	1.68	True	72.00	1.68
D	g	21	H2	True	False	-	-	True	25.00	1.03	True	53.33	2.63
A	d	22	H1	True	True	50.00	1.67	True	22.22	1.62	True	46.67	1.60
A	d	22	H2	True	True	68.42	1.40	True	94.12	0.88	True	91.43	0.88
D	a	23	H1	False	False	-	-	False	-	-	False	-	-
D	a	23	H2	False	False	-	-	True	5.26	2.06	True	5.26	2.06
C	k	24	H1	False	True	52.17	2.24	False	-	-	True	50.00	2.24
C	k	24	H2	False	False	-	-	False	-	-	False	-	-
B	g	25	H1	True	True	42.86	1.31	False	-	-	True	28.57	1.40
B	g	25	H2	True	True	95.24	0.27	True	44.44	2.25	True	50.00	2.14
A	b	26	H1	True	False	-	-	False	-	-	False	-	-
A	b	26	H2	True	False	-	-	False	-	-	False	-	-
C	i	27	H1	True	True	100.00	0.19	True	61.54	2.08	True	70.59	2.20
C	i	27	H2	True	True	55.56	2.04	True	43.33	2.59	True	43.33	2.67
A	e	28	H1	True	True	96.67	0.36	True	82.35	0.69	True	82.76	1.44
A	e	28	H2	True	False	-	-	False	-	-	False	-	-
B	h	29	H1	True	True	53.57	2.37	True	42.86	2.39	True	50.00	2.34
B	h	29	H2	True	True	7.14	1.23	True	76.92	0.55	True	7.14	1.30
D	j	30	H1	True	False	-	-	True	55.56	2.56	True	65.71	2.77
D	j	30	H2	True	True	30.00	2.49	False	-	-	True	39.29	2.67
C	b	31	H1	False	True	8.33	1.99	False	-	-	True	8.33	1.99
C	b	31	H2	False	True	10.00	1.64	True	55.56	1.87	True	53.57	1.87
A	f	32	H1	True	True	59.26	1.65	True	60.00	1.67	True	61.54	1.58
A	f	32	H2	True	True	87.10	0.58	True	53.33	1.52	True	36.67	2.07

Table D.15: Raw data table (part 2) - Goal: Balls

Ball Config	Human Config	Experiment Number	Human Name	Crossing	No Data Sharing						Data Sharing		
					R1			R2			R1+R2		
					C_h [-]	$T_{\%}$ [%]	d_G [m]	C_h [-]	$T_{\%}$ [%]	d_G [m]	C_h [-]	$T_{\%}$ [%]	d_G [m]
C	l	33	H1	True	False	-	-	True	77.78	2.35	True	77.78	2.35
C	l	33	H2	True	False	-	-	False	-	-	True	51.47	0.26
D	m	34	H1	False	True	100.00	0.19	False	-	-	True	100.00	0.19
D	m	34	H2	False	True	70.00	2.41	False	-	-	True	72.50	2.37
A	c	35	H1	False	False	-	-	False	-	-	False	-	-
A	c	35	H2	False	True	100.00	0.23	True	5.88	1.81	True	25.00	1.77
B	f	36	H1	True	True	77.14	1.94	False	-	-	True	78.95	1.94
B	f	36	H2	True	False	-	-	True	74.42	1.73	True	74.42	1.97
C	j	37	H1	False	True	97.22	0.31	True	19.23	2.60	True	40.00	2.75
C	j	37	H2	False	True	20.00	2.47	True	40.00	2.56	True	37.50	2.56
B	a	38	H1	True	True	65.38	2.04	True	64.29	1.45	True	61.54	2.08
B	a	38	H2	True	False	-	-	False	-	-	False	-	-
D	j	39	H1	False	True	100.00	0.22	False	-	-	True	100.00	0.22
D	j	39	H2	False	False	-	-	True	6.67	1.61	True	6.67	1.61
A	a	40	H1	True	True	65.52	1.85	False	-	-	True	62.07	1.93
A	a	40	H2	True	False	-	-	True	12.50	2.58	False	-	-
A	e	41	H1	False	True	75.86	1.50	True	25.00	0.32	True	78.57	1.50
A	e	41	H2	False	False	-	-	True	5.56	1.76	True	5.56	1.76
C	l	42	H1	True	False	-	-	True	65.79	2.17	True	93.62	0.95
C	l	42	H2	True	False	-	-	True	100.00	0.16	True	100.00	0.16
A	d	43	H1	False	False	-	-	False	-	-	True	57.14	1.39
A	d	43	H2	False	False	-	-	True	4.76	2.16	True	3.45	2.16
B	a	44	H1	False	True	10.00	1.49	False	-	-	True	10.00	1.49
B	a	44	H2	False	False	-	-	False	-	-	False	-	-
C	i	45	H1	False	False	-	-	False	-	-	True	100.00	0.23
C	i	45	H2	False	True	10.00	1.23	True	30.00	2.19	True	7.14	2.19
B	h	46	H1	True	False	-	-	False	-	-	False	-	-
B	h	46	H2	True	False	-	-	False	-	-	False	-	-
D	m	47	H1	False	True	83.33	0.34	False	-	-	True	83.33	0.34
D	m	47	H2	False	True	12.50	0.46	True	20.00	0.53	True	14.29	0.48
B	f	48	H1	True	True	4.17	1.72	False	-	-	True	4.17	1.59
B	f	48	H2	True	False	-	-	True	59.46	2.28	True	58.97	2.52
D	g	49	H1	True	True	36.67	2.53	False	-	-	True	33.33	2.57
D	g	49	H2	True	False	-	-	True	10.00	0.97	True	10.00	1.14
A	c	50	H1	False	True	40.91	2.30	False	-	-	True	40.91	2.30
A	c	50	H2	False	False	-	-	False	-	-	False	-	-
C	j	51	H1	True	True	100.00	0.13	False	-	-	True	100.00	0.13
C	j	51	H2	True	True	11.11	2.60	True	23.53	2.54	True	15.00	2.67
D	a	52	H1	False	True	53.85	0.61	False	-	-	True	53.85	0.61
D	a	52	H2	False	True	70.00	2.11	False	-	-	True	86.36	2.11
A	f	54	H1	True	True	100.00	1.81	False	-	-	False	-	-
A	f	54	H2	True	True	76.92	1.05	True	55.17	1.82	True	64.52	1.17
A	b	56	H1	True	False	-	-	False	-	-	False	-	-
A	b	56	H2	True	False	-	-	True	88.89	1.05	True	36.36	1.15
B	h	57	H1	True	False	-	-	False	-	-	False	-	-
B	h	57	H2	True	False	-	-	False	-	-	False	-	-
C	b	58	H1	False	False	-	-	False	-	-	True	90.70	1.02
C	b	58	H2	False	False	-	-	False	-	-	False	-	-
A	d	59	H1	False	True	3.85	2.02	False	-	-	True	76.92	0.64
A	d	59	H2	False	True	30.00	1.46	True	77.78	1.28	True	70.00	1.28
D	g	60	H1	True	False	-	-	False	-	-	False	-	-
D	g	60	H2	True	True	100.00	0.08	True	76.47	2.59	True	73.58	2.95
A	f	61	H1	True	True	20.00	1.72	True	21.05	1.73	True	4.55	1.73
A	f	61	H2	True	True	42.31	2.10	True	28.57	1.81	True	7.06	2.23
D	a	62	H1	True	True	78.38	2.05	True	64.52	2.38	True	64.86	2.78
D	a	62	H2	True	True	50.00	2.45	False	-	-	True	57.78	2.92
A	e	63	H1	False	True	66.67	1.85	False	-	-	False	-	-
A	e	63	H2	False	True	-	-	True	54.17	1.79	True	17.81	1.79
C	i	64	H1	False	False	-	-	False	-	-	True	100.00	0.13
C	i	64	H2	False	False	-	-	False	-	-	False	-	-
A	b	65	H1	True	True	50.00	1.77	True	7.14	2.04	True	47.37	1.60
A	b	65	H2	True	True	43.75	1.67	False	-	-	True	58.33	2.13
B	g	66	H1	True	True	100.00	0.24	True	51.43	2.49	True	47.37	2.66
B	g	66	H2	True	False	-	-	False	-	-	False	-	-
A	c	67	H1	False	False	-	-	False	-	-	False	-	-
A	c	67	H2	False	True	46.67	1.50	True	80.00	1.42	True	80.43	1.48

Table D.16: Raw data table (part 3) - Goal: Balls

Ball Config	Human Config	Experiment Number	Human Name	Crossing	No Data Sharing						Data Sharing		
					R1			R2			R1+R2		
					C_h [-]	$T_{\%}$ [%]	d_G [m]	C_h [-]	$T_{\%}$ [%]	d_G [m]	C_h [-]	$T_{\%}$ [%]	d_G [m]
C	k	68	H1	False	False	-	-	True	66.67	1.48	True	68.00	1.73
C	k	68	H2	False	False	-	-	True	38.46	2.32	True	37.84	2.84
D	j	69	H1	True	False	-	-	False	-	-	False	-	-
D	j	69	H2	True	False	-	-	True	64.10	2.36	True	71.79	2.31
B	a	71	H1	False	True	14.29	1.62	False	-	-	True	14.29	1.62
B	a	71	H2	False	True	33.33	0.35	True	5.88	1.99	True	5.56	1.99
D	m	72	H1	True	True	58.62	2.18	False	-	-	True	62.50	2.18
D	m	72	H2	True	True	57.14	2.49	True	96.88	0.28	True	42.11	2.94
C	b	73	H1	True	False	-	-	True	71.43	1.49	True	65.52	2.05
C	b	73	H2	True	True	100.00	0.10	True	96.15	0.26	True	45.83	1.44
B	f	74	H1	True	False	-	-	True	43.75	2.32	True	48.48	2.44
B	f	74	H2	True	False	-	-	False	-	-	False	-	-
C	l	75	H1	True	True	47.62	2.13	True	97.37	0.26	True	55.26	2.56
C	l	75	H2	True	True	64.71	1.60	False	-	-	True	72.73	1.92
D	g	76	H1	False	True	4.00	2.42	False	-	-	True	4.00	2.45
D	g	76	H2	False	False	-	-	True	4.55	2.27	True	4.17	2.37
C	j	77	H1	False	False	-	-	False	-	-	False	-	-
C	j	77	H2	False	True	7.14	2.41	False	-	-	True	33.33	2.41
A	d	78	H1	False	True	9.09	1.41	True	-	0.25	True	11.11	1.41
A	d	78	H2	False	False	-	-	True	71.43	1.48	True	71.43	1.48
C	i	79	H1	False	True	43.75	2.55	True	17.39	2.60	True	29.03	2.87
C	i	79	H2	False	True	11.11	2.57	False	-	-	True	33.33	2.48
A	b	80	H1	False	True	74.07	1.48	False	-	-	True	66.67	1.48
A	b	80	H2	False	False	-	-	True	8.33	1.58	True	8.33	1.58
D	a	81	H1	True	False	-	-	False	-	-	False	-	-
D	a	81	H2	True	False	-	-	False	-	-	True	35.56	0.82
C	b	82	H1	True	True	70.00	1.87	False	-	-	True	24.32	2.12
C	b	82	H2	True	True	66.67	1.72	False	-	-	True	82.93	1.72
B	f	83	H1	True	True	76.60	2.40	False	-	-	True	39.78	2.55
B	f	83	H2	True	True	87.88	1.06	False	-	-	True	88.57	1.06
C	k	84	H1	False	False	-	-	False	-	-	False	-	-
C	k	84	H2	False	False	-	-	False	-	-	False	-	-
B	a	85	H1	False	True	3.85	1.94	False	-	-	True	1.27	1.94
B	a	85	H2	False	True	50.00	2.47	False	-	-	True	64.58	2.85
A	e	86	H1	False	False	-	-	False	-	-	False	-	-
A	e	86	H2	False	False	-	-	False	-	-	False	-	-
B	h	87	H1	True	True	31.25	1.51	False	-	-	True	6.25	1.36
B	h	87	H2	True	False	-	-	False	-	-	False	-	-
A	f	88	H1	True	True	26.67	1.63	False	-	-	True	46.81	2.37
A	f	88	H2	True	False	-	-	False	-	-	False	-	-
B	g	89	H1	False	True	59.38	1.38	True	40.74	1.52	True	41.38	1.53
B	g	89	H2	False	False	-	-	True	5.26	1.84	True	5.26	1.90
A	a	90	H1	True	True	65.62	1.89	True	81.25	0.75	True	71.43	1.40
A	a	90	H2	True	False	-	-	False	-	-	False	-	-
A	c	91	H1	False	False	-	-	True	45.45	1.27	True	89.66	0.75
A	c	91	H2	False	True	8.33	2.13	True	31.58	2.50	True	24.00	2.50
C	l	92	H1	True	False	-	-	True	82.76	1.09	True	68.57	1.81
C	l	92	H2	True	True	53.33	1.92	True	66.67	1.98	True	63.64	2.38
D	j	93	H1	True	False	-	-	True	82.76	1.11	True	79.07	2.45
D	j	93	H2	True	False	-	-	True	94.74	0.26	True	94.74	0.26
C	i	94	H1	False	False	-	-	False	-	-	False	-	-
C	i	94	H2	False	False	-	-	True	85.71	0.51	False	-	-
D	m	95	H1	True	False	-	-	True	68.97	2.08	True	70.00	2.20
D	m	95	H2	True	True	68.97	2.11	False	-	-	True	34.48	3.03
C	j	96	H1	True	False	-	-	False	-	-	False	-	-
C	j	96	H2	True	True	63.64	1.58	True	78.38	1.44	True	76.32	1.59
A	d	97	H1	False	True	72.73	1.35	True	46.15	1.67	True	57.14	1.97
A	d	97	H2	False	False	-	-	True	71.43	1.59	False	-	-
C	b	98	H1	False	False	-	-	False	-	-	False	-	-
C	b	98	H2	False	True	11.11	1.56	True	47.37	1.88	True	39.13	1.88
B	g	99	H1	False	True	3.70	2.10	False	-	-	True	3.70	2.08
B	g	99	H2	False	True	68.42	0.69	True	10.00	1.43	True	8.33	1.59
C	k	100	H1	True	False	-	-	False	-	-	False	-	-
C	k	100	H2	True	False	-	-	False	-	-	False	-	-
A	b	101	H1	False	False	-	-	False	-	-	False	-	-
A	b	101	H2	False	False	-	-	False	-	-	True	-	0.11

APPENDIX E
CONSENT FORM

The consent form signed by all participants before the experiments.

CONSENT FORM FOR PARTICIPATION IN RESEARCH

The purpose of this form is to inform you of the nature of the experiment that is about to start and to ask for your informed consent to participate.

Your participation in this experiment is strictly voluntary.

You may choose not to participate, and you may withdraw at any time during the experiment.

If there are any hesitations about participating or you have any questions about the research, please feel free to ask the experimenter, Sander van der Vorst.

You will be asked to pick-up balls and bring them in front of cameras (see experiment introduction paper). You will be recorded while moving on the field. There will be no health risks. The video of this experiment will be recorded and notes will be taken. All data obtained in this experiment will be processed and reported anonymously. More information about this study has been explained to you by the researcher.

I, (NAME)..... have read and understood this consent form and have been given the opportunity to ask questions. I agree that I participate in research mentioned above.

Participant Signature and Date

APPENDIX F
EXPERIMENT INFORMATION

A form containing all relevant experiment information for the participants. It was distributed to the participants before the start of their experiments.



Learning in Mixed Human-Robot Teams

DATA GATHERING EXPERIMENT

Dear Participant,

First of all, thank you for your willingness to participate in one of the experiments that I am conducting for my master thesis. In this document the details of the experiment are briefly explained. If you have any further questions, please do not hesitate to ask them.

Kind regards,
Sander van der Vorst

1 Goal of Experiment

The goal for this experiment is to simultaneously record two sequences of two persons each walking towards a different soccer ball, picking it up and delivering it to a robot.

2 Experiment Setup

The experiment will be set up as follows:

- Both persons start at a designated start location, which will be different for every recording.
- Both persons will walk towards a different ball. Your path does not have to be a straight line but take the path you want (without exiting this half of field).
- When arriving at a ball, the ball has to be picked up. Both persons do not have to do this necessarily at the same time.
- When a person has picked up a ball, he/she walks towards a robot. Both persons cannot arrive at the same robot at the same time.
- When arriving at a robot, the ball can be put on the floor on the middle line of the field before the robot (the spot will be marked).
- When the ball has been laid down, the person can exit the field at the closest field corner of the half field.

During the experiment it is advised not to walk too fast but also not overly slow, so the camera's can later track all movement correctly.

3 Time

One sequence will be filmed in \pm 1 minute. Then the balls will be relocated and the camera's restarted, which will take a few minutes. The total time of all experiments will be 1 hour at maximum.