Planning

Week 5:

Make individual planning (Done)

Task environment description (Done)

Finalising user requirements through proxemics (Done)

Looking at viability of Dynamic window approach and Social Force model. (Done)

Per approach: Try to understand it, give it a basic description, look at what user requirements are satisfied, if and how it might be extended / adapted for this application. (Done)

Decide on the best approach. (Done)

Week 6:

Find a way to simulate SFM through MATLAB (Done)

Quantify user requirements if possible (Done)

Finish performance calculations for simulation for testing user requirements (Done)

  Implementation in pseudo-code (Done)

Give extension with face pose (Done)

Cost function object padding + occlusion (Done)

Week 7:

Finish conclusion + discussion

Write discussion

  Describe why adapted SFM is thought to be superior

Topics for further research

Working on final presentation

Finalising report

  Add table of contents

  Page numbering

  Fix references

  Spelling & grammar check

Week 8:

22-10-2018 Final Presentation

User-centred design of a collision avoidance procedure for robots in supermarket environments

Introduction & problem statement

Robot navigation and collision avoidance in crowded and dynamic environments is a challenging problem, not only from a technical point of view, but also when looking at how robots should behave in the proximity of (large numbers of) people. For various (social) robot applications, finding a method that is desirable for humans while maintaining efficiency is of great importance. Robot applications in supermarkets is rising in popularity, examples being Best Buy's *Chloe* robot [BRON][1] and Simbe Robotics' Tally robot [BRON][2] so it would be useful to develop methods that are specifically designed for a supermarket environment keeping customer and staff requirements in mind.

This research will focus on finding a solution for robot collision avoidance in a supermarket environment. A supermarket environment has aspects that make it unique from other crowded environments. To make this more concise, a description of this environment is given with advantages and difficulties for designing a robot collision avoidance. Furthermore, it will also become clear that users (staff and customers) will have certain requirements that relate to human robot interactions (HRI). Keeping both the environment and user requirements in mind, two state-of-the art collision avoidance procedures will be assessed on application in a supermarket environment and possible additions to enhance them for this application will be investigated. A simulation with a candidate object avoidance procedure will be done to test its working potential. A description on how this procedure can be tested in simulation to satisfy user requirements is given and adaptations to a collision avoidance approach are presented. A motivation is given on why the adapted approach is the approach for supermarket collision avoidance and finally the need for actual simulation and further research regarding this approach is described.

Task environment description of a supermarket

We will look at advantages and difficulties for robot collision avoidance in supermarkets. Through this analysis several aspects that need investigation will come to light.

Advantages for collision avoidance

1. It is assumed that there are several (security) cameras already mounted on the ceiling and that the robot already possesses an omnidirectional camera. By giving the robot access to ceiling mounted cameras, these can be used for collision avoidance as extra sensory input on top of the camera already present on the robot itself. This gives the robot a top down view of the area he is in, filling in blank spots in the robot's local sensing. This poses several questions; for one, security cameras usually make use of fish-eye cameras giving a distorted view of the environment, meaning that these images might need to be processed or not usable at all. Then also, how many extra ceiling cameras would be necessary and how much would that cost?

---

[1] Robots in Retail - Examples of Real Industry Applications -. (2017, July 06). Retrieved from https://www.techemergence.com/robots-in-retail-examples

[2] Tally shelf-scanning robot adds RFID, machine learning. (2018, July 20). Retrieved from https://www.therobotreport.com/tally-robot-rfid-machine-learning

Takaaki Sato et al. [BRON] [3] have proved that fish eye cameras can be used to make a (2D) bird's eye view of an environment to eliminate blind spots in a robot's local sensing. However, it needs to be investigated whether it is still desirable for a supermarket enterprise to invest in more cameras, when the cheaper option of only using local robot cameras might suffice.

2. Supermarket aisles have a static layout, with each aisle having distinct retail products ordered in a known layout. This semantic information stored in retail products can be used for robot localisation and navigation from point A to B. A detailed description of navigation using semantic techniques is given by Cosgun and Christensen [BRON][4]. Since this is not applicable for collision avoidance, it will not be discussed here. However, this a promising starting point for research in navigation algorithms for supermarkets.

Difficulties for collision avoidance

1. Customers and staff members will be walking around supermarkets, either in groups or alone, maybe carrying a shopping cart. All these people need to be avoided in a way that is perceived as safe by them. The robot should therefore act differently when humans, instead of (static) inanimate objects are to be avoided. To find out how a robot should act differently among humans, an investigation on proxemics for HRI needs to be done.

2. There are also peak times in number of customers walking around (e.g. on Saturdays). Collision avoidance procedures on their own might then lead to the robot having no way to avoid masses of people or lead to computationally expensive situations where the robot loses reactiveness. A solution for robot collision avoidance in masses might be found when looking at how humans tend to cooperate to avoid each other in these situations. Conventional collision avoidance procedures might tell the robot to not come closer than X metres to humans to respect their personal space. However, humans sometimes tend to avoid each other crossing this line in crowded situations still maintaining to be polite and/or comfortable with each other. This is something that needs to be considered in collision avoidance procedures. Moreover, procedures might need to be adapted so that crowded areas are detected and then treated in a more computationally light way. In this situation it might also be necessary to add visible or audible cues that alert surrounding customers in a comfortable way to make sure the robot is noticed by surrounding humans to facilitate movement in crowded spaces. It should be investigated what kind of cues are desirable in these situations, how (computationally) inefficient some procedures become when large groups of people are in the robot's vicinity and how these inefficiencies can be overcome.

3. (Parked) shopping carts are present, which are objects that can move but not necessarily. For a parked shopping cart case, there should be some prediction about probability that it will move and in what direction. This probability should be depended on whether a human is close to that cart. These probabilities might be incorporated in a cost function for shopping carts specifically. It needs to be investigated if this motion prediction is worth the extra computational cost to the algorithm, possibly by doing simulations in combination with real-life experiments.

---

[3] Sato, T., Moro, A., Sugahara, A., Tasaki, T., Yamashita, A., & Asama, H. (2013). Spatio-temporal bird's-eye view images using multiple fish-eye cameras. Proceedings of the 2013 IEEE/SICE International Symposium on System Integration, 753–758. doi: 10.1109/SII.2013.6776674

[4] Akansel Cosgun, Henrik I. Christensen. (2018). Context-aware robot navigation using interactively built semantic maps. De Gruyter Open. Paladyn, J. Behav. Robot. 2018

4. Miscellaneous items such as boxes, pallets or retail products fallen from shelves might be present as obstacles. Ceiling mounted cameras should be able to detect these obstacles. Since these objects are static, no movement prediction is necessary. The location of static objects can be sent to the robot directly or can be sensed by the robot itself and path planning can be adapted accordingly.

Identifying user requirements

Proxemics and HRI

In order to find a desirable way in which robots avoid and move alongside customers and staff members, user requirements will be looked at.

The term collision avoidance in general will be used for the avoidance of all entities in a supermarket, being: humans, moving objects and stationary objects. When avoiding or moving close to humans, it is important that humans do not feel any discomfort, harm or surprise. To make these and related terms more concise the definitions of Thibault, K et al. **[BRON]**[5] will be used:

Comfort is the absence of annoyance and stress for humans in interaction with robots.

It should be noted that comfort is different than safety, in that a robot can move about safely but the surrounding people may feel unsafe. The opposite is also possible, when the human perceives a robot moving about safely it can still end up in a collision.

Naturalness is the similarity between robots and humans in low-level behaviour patterns.

Naturalness thus strives to a physical imitation of humans as much as possible. Examples are movement speeds and robot shapes that resemble humans.

Sociability is the adherence to explicit high-level cultural conventions.

Sociability is seen as constraints posed by society. Examples are the rule to walk on the righthand side and politely asking someone to move out of the way.

Several robot user requirements for avoiding collision with customers will be looked at now. Most of them come from surveys presented by the literature summary of Thibault, K et al. and from studies in the field of proxemics.

1. Robots should never come too close to humans, even during object avoidance routines. It could frighten humans, possibly leading to sudden actions and human injury.

E. Hall **[BRON]**[6] found designations for interpersonal distances for several human to human interactions:

[5] Thibault Kruse, Amit Pandey, Rachid Alami, Alexandra Kirsch. Human-Aware Robot Navigation: A Survey. Robotics and Autonomous Systems, Elsevier, 2013, 61 (12), pp.1726-1743.
[6] E. Hall. The hidden dimension. Anchor Books, 1966.

Designation	Specification	Reserved for ...
Intimate distance	0 - 45cm	Embracing, touching, whispering
Personal distance	45 - 120cm	Friends
Social distance	1.2 - 3.6m	Acquaintances and strangers
Public distance	> 3.6m	Public speaking

Figure 1 proxemics table from E. Hall

This table can be used to find a proper distance for robots during an avoidance or general movement that respects the personal zones of people. Generally, to make a person feel safe the robot should try to avoid the intimate and personal space of people, so a distance of more than 120 cm would be preferred during avoidance. Although this table does not incorporate the fact that a robot instead of a human is entering these personal spaces, current research still suggests that using these distances as a basis for robot navigation and collision avoidance is still a viable option. As stated earlier, however, this distance of 120 cm should not be implemented as a no-go zone for robots. If the robot is noticed by surrounding humans and has adapted its speed accordingly to some desirable approaching speed, it should be able to enter person's intimate distance temporarily for collision avoidance.

2. Robots should not block a human's path, which may cause frustration.

This requirement is rather straightforward, however, Thibault, K et al. describe that when humans actively try to avoid robots as well (when the robot's movement is perceived as safe and predictable) this is not necessarily a problem anymore. This requires that the robot is easily noticed by surrounding people.

3. In a case of a densely crowded area, the robot should provide humans with a visible or audible cue, possibly through language, to make collision avoidance possible or easier.

In this situation a robot should make clear to surrounding people that it wants to move in a certain direction and that some people might need to adapt their (walking) behaviour accordingly by, for example, making way for the robot. This cue should be as effective as possible in crowded environments, while also making sure that people do not lose comfort. The robot will thus attempt to persuade people to change their actions, which some people may dislike or even try to resist against. Ghazali, A. S. Et al [BRON][7] describe that this phenomenon is described as psychological reactance, which can result from people perceiving these persuasive attempts as threats towards their freedom in decision making. This research concludes that using highly controlling language can lead to successful persuasions and that the cues presented by the persuading robot such as facial expressions do not have an impact. High controlling language uses explicit verbs, an example could be "You must..." etc. For this purpose of collision avoidance in a supermarket, it is thought that a verbal output using low controlling language, thus for example, "Could you please make way for

[7] Ghazali, A. S., Ham, J., Barakova, E. I., & Markopoulos, P. (2017). Pardon the rude robot: Social cues diminish reactance to high controlling language. 2017 26th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN), 411–417. doi: 10.1109/ROMAN.2017.8172335

me?" or something similar, is desirable to be used. According to Lohse, M. Et al. [BRON][8] People tend to have a positive attitude towards robots that communicate its intention through audio, compared to robots that do not communicate its intentions. Therefore, this is the preferred way of interacting with customers in crowded areas. However, it will still be necessary to do further research on which exact sentences using low controlling language are the most effective and fitting in for this context, minimising social reactance.

4. Robots should not move/approach too fast, which leads to discomfort for surrounding people.

Butler and Agah **[BRON]**[9] found that approaching with 1 [m/s] turned out uncomfortable, while 0.5 [m/s] was acceptable. During avoidance the situation is slightly different, but the same velocities could be used. An important aspect of robot movement is the degree in which it is predictable, understandable or readable for humans (natural). According to Hayashi **[BRON]**[10] and Satake **[BRON]**[11] a speed that adapts to or resembles surrounding humans would be desirable for general movement. Humans tend to have a preferred walking speed of 1.4 [m/s] [BRON][12], so it is thought that the robot's speed should always be lower than that, while a velocity of 0.5 [m/s] is admissible during the event that the robot (temporarily) enters someone's intimate space for collision avoidance.

5. Avoid erratic motions during movement, especially when close to humans.

This refers to the aspect of smoothness, which means that the geometry of the taken path and the velocity profile should be smooth. This would improve the naturalness of robots. To ensure a smooth velocity profile that resembles humans, it is necessary to impose constraints on the acceleration of the robot. Human pedestrian acceleration is found to have a maximum value of 1.44 [m/s^2] with an average of 0.68 [m/s^2] [BRON][13]. The average acceleration of a human should then be set as a maximum for robot acceleration.

6. Robots should not make noises that cause distraction when coming close to humans, to increase comfort.

Comfortable robot motion should also pose a constraint on robot noise. Lohse, M et al. Did experiments where approach speed and sound volume of a robot were chosen according to four situations depicted in the figure below:

---

[8] M. Lohse, N. Van Berkel, E. Van Dijk, M. Joosse, D. Karreman, V. Evers. The influence of approach speed and functional noise on users' perception of a robot - IEEE Conference Publication. (2018, October 13). Retrieved from https://ieeexplore.ieee.org/abstract/document/6696573

[9] J. T. Butler and A. Agah. Psychological effects of behavior patterns of a mobile personal robot. Autonomous Robots, 10(2):185–202, 2001.

[10] K. Hayashi, M. Shiomi, T. Kanda, and N. Hagita. Friendly patrolling: A model of natural encounters. In Proceedings of Robotics: Science and Systems, Los Angeles, CA, USA, June 2011.

[11] D. F. G. M. I. H. I. N. H. S. Satake, T. Kanda. How to approach humans? strategies for social robots to initiate interaction. In HRI, ACM/IEEE, 2009.

[12] Browning, R. C., Baker, E. A., Herron, J. A., & Kram, R. (2006). Effects of obesity and sex on the energetic cost and preferred speed of walking. Journal of Applied Physiology. Retrieved from https://www.physiology.org/doi/full/10.1152/japplphysiol.00767.2005

[13] Korhonen, T., & Heliövaara, S. (2011). FDS+Evac: Modelling Pedestrian Movement in Crowds. SpringerLink, 823–826. doi: 10.1007/978-1-4419-9725-8_81

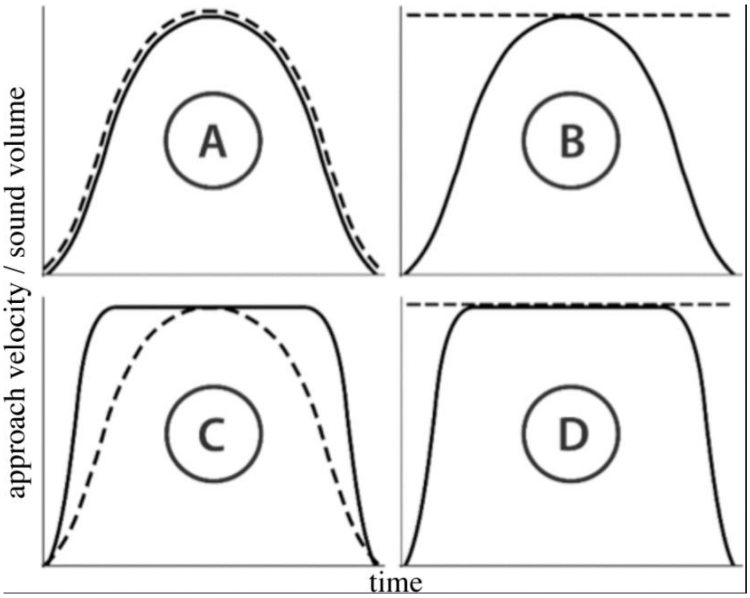

Figure 2 Four situations showing sound level (dashed line) and approaching velocity (solid line) profile of a robot, from Lohse, M. Et al.

It was found that robot sound volume has an impact on the perceived velocity of an approaching robot. They conclude that people tend to prefer a sound profile that almost matches the velocity profile (situation C), if we look at perceived safety. If we look at the likeability of a robot, sound profile A was preferred. In conclusion, no values of loudness were found in this research, but it is safe to say that some form of sound that scales with the velocity is preferred and would increase comfort levels for surrounding humans.

7. Behaviours disliked by society and the dominant culture should be avoided.

As described by Thibault, K et al., robots might need to prefer one side of the aisle for movement and/or avoidance, depending on country and culture. The robot might also need to ask or give cues to its environment if it wants to avoid a human or notices a human is blocking its path (as described under requirement 3). These aspects would make the robot more sociable.

Describing cost functions

The most straightforward way to implement these user requirements is by making use of cost functions that can be implemented in avoidance procedures.

In order to find a path avoiding a human, in a sufficiently safe, comfortable, natural and legible way, a cost function is used. This cost function assigns cost values to robot actions, depending mostly on environment and the robot's state. This cost function can be expanded to the environment's geometry, type and state, the person's age and gender, their current activity, the current interactions between people and interactions between people and objects. All this knowledge it has about its environment is stored in this cost function, which it tries to minimise when choosing a way to avoid collision. A visualisation of several cost function as a 2D map is seen in the following figure

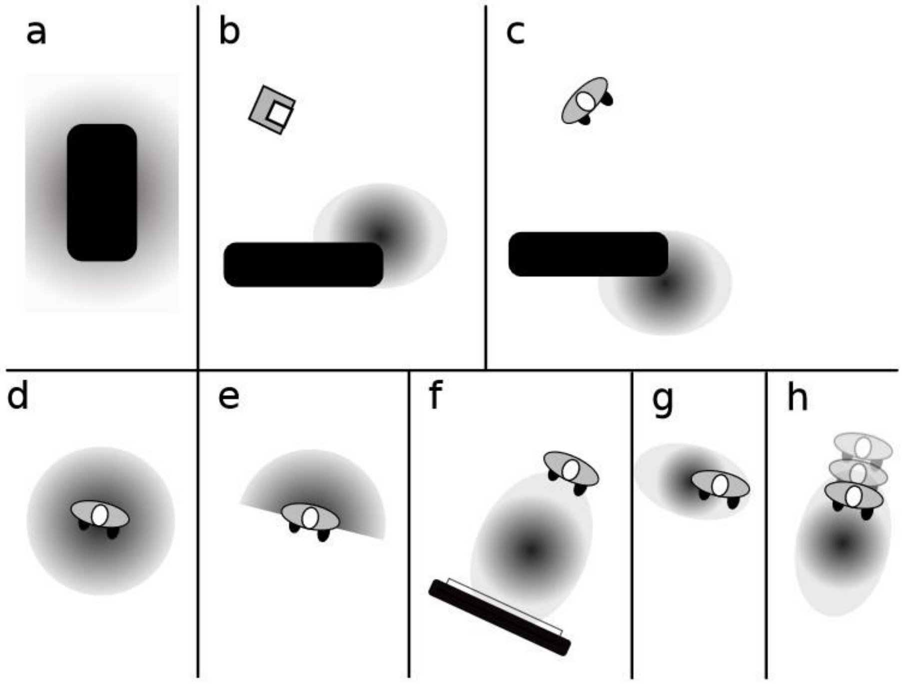

Figure 9: Visualizations of separate costmaps, thick black areas are obstacles, the square in b is a robot, human in h is moving. Areas shaded in grey have costs, meaning the robot should prefer to avoid those if possible. Shapes and sizes of cost function may vary and be context-dependent. All cost-functions can be combined.

Figure 3 visualisation of cost functions from Thibault, K et al.

Cost functions can incorporate the user requirements stated previously, by for example, modelling that moving closer to humans has less cost if done at low speed than at high speed. A problem with this might be that search space increases tremendously, resulting in a loss of robot reactiveness. A combination of the following cost functions is thought to be appropriate for a supermarket environment.

- Object padding (seen in figure 3a)

Object padding can be useful so that the robot does not move too close to supermarket shelves, possibly causing misaligned products to fall out of shelves creating further complexities.

- Object occlusion and hidden zones (seen in figure 3b and c combined)

Due to the chaotic nature of the environment, people can come rushing around corners leading to possible collisions with robots that are just behind line of sight for humans. The robot should know these locations and avoiding them is desirable.

- Basic comfort distance (seen in figure 3d)

Following the previously described user requirements, every person's personal space needs to be avoided as much as possible. An example of a procedure that incorporates this is given by Barnaud, M.-L et al. [BRON][14] who proposed a model that maps this personal space on the environment through a 2D normal distribution as a cost function, which can be used for collision avoidance. It was also found that interaction space, being the space in between two humans conversing or interacting in some way, was not necessary to model for these procedures. This model was successfully

[14] Barnaud, M.-L., Morgado, N., Palluel-Germain, R., Diard, J., & Spalanzani, A. (2014, September 14). Proxemics models for human-aware navigation in robotics: Grounding interaction and personal space models in experimental data from psychology. Retrieved from https://hal.archives-ouvertes.fr/hal-01082517

validated with experimental results with an actual robot. It showed that these procedures were perceived as safe by humans while also maintaining efficiency.

- Passing people on their left (seen in figure 3g)

Passing people on their left is a social convention that should be preferred by the robot during collision avoidance. This is mostly a convention when a person avoids someone from behind. During face to face interactions, people tend to look in the direction they want to go. In these situations, this information should be used instead for collision avoidance as it is perceived natural for humans.

- Space ahead for moving (seen in figure 3h)

In general, robots should avoid moving in this space, as it hinders people. This does require some form of motion prediction.

Most cost functions have growing costs as the distance to some entity decreases. This can of course be tweaked to exponential or other functions. For this application it is probably not necessary to change this parameter. Combining these cost functions can be done via weighted sums. Cost function shape, combination and weighting can be tweaked manually or through machine learning.

Distinguishing between humans and objects

This distinction is needed, because humans will be avoided in a more advanced way than moving or static objects. This can be achieved through object recognition; however, this concept is beyond the scope of this research. Neglecting this aspect will make it so that only a distinction between moving and static objects will have to be made by sensors. By avoiding all moving objects in the same way as humans would be avoided, the main problem is slightly simplified. For real world applications this distinction can of course not be neglected, but the velocity detection discussed in the next section can easily be extended with the recognition of human beings.

Distinguishing between (possibly) moving and static objects

The robot can make this distinction through object recognition as shown by Wei, Z et al. [BRON][15]. This approach makes use of feature-line flows and distinguishes moving from static objects by computing residual errors. Although it will not be discussed here, research on how object recognition for navigation and collision avoidance might be used in supermarket environments is very important.

Assessment of possible collision avoidance procedures

Collision avoidance procedures will now be assessed on their application in a supermarket environment. Initially, the main aspects of the algorithm are described, then the degree in which these approaches can satisfy user requirements is looked at. Finally, a conclusion is drawn on how this approach might need to be adapted or extended to better fit the environment.

The environment of a supermarket is for this assessment simplified to one aisle that the robot needs to navigate through, this is done to fully leave out the navigation aspect for robots in the environment. During its path it will encounter static objects, moving objects and several humans standing around, walking and interacting the robot all needs to avoid in a reactive manner. Furthermore, it is assumed that a top-down view of the aisle is accessible to the robot by using

[15] Wei, Z., Zhu, H., & Wang, P. (2007). An Object Recognition Method for Indoor Robot Based on Feature-Line Flows. 2007 IEEE International Conference on Automation and Logistics, 591–596. doi: 10.1109/ICAL.2007.4338633

images of several (fish-eye security) cameras mounted on the ceiling. This environment is illustrated in the following figure.

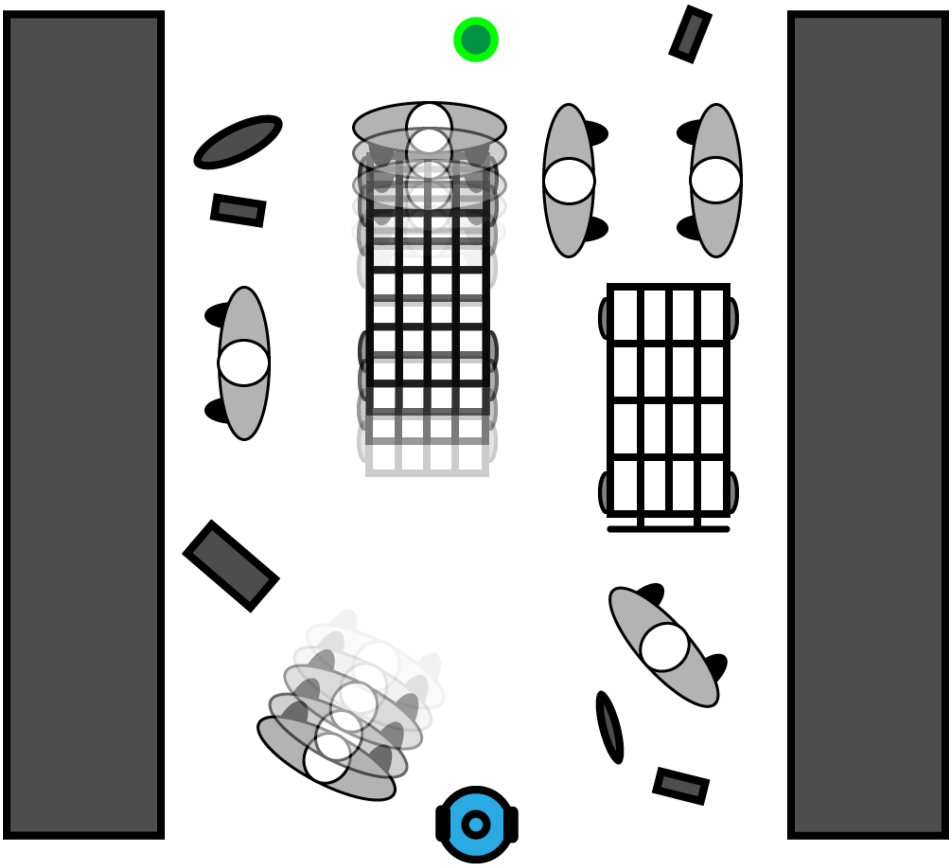

Figure 4 a schematic overview of the collision avoidance environment, the robot is represented in blue while the goal is represented in green.

Dynamic window approach

The dynamic window approach by Fox, D. et al. [BRON][16] will be discussed

The dynamic window approach describes robot motion directly in the space of velocities. It reduces the search space to a dynamic window, which consists of the velocities reachable within a short time interval. These velocities are only admissible if the robot is also able to stop completely and safely in this time-span. It makes use of an objective function which measures the progress towards a goal location, forward velocity and distance to the next obstacle on the trajectory.

This approach models velocity as a piecewise constant function in time. It is thus assumed that robot trajectories consist of finitely many segments of circles. Intersection between circles and obstacles are used for collision checking. The approximate motion equations for x and y coordinates are described as follows:

[16] Fox, D., Burgard, W., & Thrun, S. (1997). The dynamic window approach to collision avoidance. IEEE Robotics & Automation Magazine, 4(1), 23–33. doi: 10.1109/100.580977

For the x coordinate:

$$x(t_n) = x(t_0) + \sum_{i=0}^{n-1} \left(F_x^i(t_{i+1})\right)$$

$$F_x^i(t) = \begin{cases} \frac{v_i}{\omega_i}(\sin\theta(t_i) - \sin(\theta(t_i) + \omega_i \cdot (t - t_i))), \, \omega_i \neq 0 \\ v_i \cos(\theta(t_i)) \cdot t, \, \omega_i = 0 \end{cases}$$

And analogously for the y coordinate:

$$y(t_n) = y(t_0) + \sum_{i=0}^{n-1} \left(F_y^i(t_{i+1})\right)$$

$$F_y^i(t) = \begin{cases} -\frac{v_i}{\omega_i}(\cos\theta(t_i) - \cos(\theta(t_i) + \omega_i \cdot (t - t_i))), \, \omega_i \neq 0 \\ v_i \sin(\theta(t_i)) \cdot t, \, \omega_i = 0 \end{cases}$$

These equations make use of a discrete set of time steps (n).

v_i is the translational velocity at timestep i

ω_i is the rotational velocity at timestep i

$\theta(t_i)$ is the global orientation of the robot

These equations only depend on velocity, but these velocities can of course not be chosen arbitrarily. They need to follow from the dynamic situation the robot is in.

The search algorithm decides what velocities are admissible, which they are if the robot is able to stop before it reaches the nearest obstacle. Also, these velocities are restricted in that only velocities that can be reached in a short time interval (the dynamic window) will be chosen.

The robot then maximises the objective function, by picking a trajectory that maximises its translational velocity and the distance to obstacles but minimizing the angle to its goal relative to its own heading direction.

The main disadvantage of this approach is that it does not consider at all what kind of obstacles are in the environment and it only assumes static objects are present. There is no distinction made between moving and static objects, but more importantly, it does not consider that humans might need to be avoided differently. This approach also does not benefit much from the use of a top-down view as this approach is purely based on local reactive planning. An advantage of this approach is that it is very explicit about its movement trajectory through the functions for x and y that only depend on translational and rotational velocities.

Because of the disadvantages, the algorithm as it is presented here is not very viable for a supermarket environment. The restricted admissible velocities that result from this approach do make sure that erratic motion of the robot is prevented. This means that only user requirement 5 is satisfied.

To make this approach more viable for a supermarket one will need to introduce the concept of moving obstacles, therefore needing an extension with motion prediction. If more user requirements are to be satisfied, this approach should be extended even more with previously described cost functions and constraints formulated in the user requirements section. Seder and Petrovic [BRON][17] describe the dynamic window approach with motion prediction. Henkel and Xu [BRON][18] describe the extension with a cost function, but this cost function has nothing to do with enhancing the human robot interaction during collision avoidance.

Social force model

The Social Force model as described by Ratsamee, P. et al. [BRON][19] will be discussed now.

This is a very promising approach, since it aims to predict human motion through calculated social forces and then uses it in robot path planning. Social forces are described as inner motivation of a person to reach a certain goal. This path planning is perceived as human-like, because its path is natural, smooth and very much predictable for other human beings in the same environment. This approach specifically also distinguishes between objects and humans by analysing people's face pose. People tend to look in the way they want to avoid a certain obstacle or other person, so this is very valuable information when an avoidance that is predictable by humans needs to be executed. So, this approach considers the physical constraints of avoiding obstacles as well as social constraints.

This approach works by calculating a resulting force, $\sum F$, for changing the motion of individual humans or robots. This resulting force is calculated from F^{goal}, an attractive force that leads the human towards his goal, F^{object}, a repulsive force from other objects and F^{human}, a repulsive force from other humans: $\sum F = F^{goal} + F^{object} + F^{human}$. F^{object} and F^{human} are then calculated from a combination of social repulsive foces, f^{social} and physical repulsive forces $f^{physical}$.

For incorporating the face pose of surrounding humans a new force is added:

$$F^{facepose} = FS \, * \, e^{\frac{r_{i,R} - d_{i,R}}{s_R}} \, * \, \overrightarrow{\boldsymbol{v_{i,R}}} \, * \, (\lambda + (1 + \lambda)\frac{1+cos(\theta)}{2}$$

In this formula, the following holds:

FS is a constant term that represents the strength of the face pose effect.

s_R is the range of the force

$d_{i,R}$ is the distance between the robot and human

$r_{i,R}$ is the sum of the radius of robot and human

$\mathbf{v_{i,R}}$ is the face pose vector from a human related to the robot. This describes the force direction.

θ describes the difference in angle between a human's face pose and the robot's.

λ is the anisotropic factor related to the cosine term in $F^{facepose}$

---

[17] Seder, M., & Petrovic, I. (2007). Dynamic window based approach to mobile robot motion control in the presence of moving obstacles. Proceedings 2007 IEEE International Conference on Robotics and Automation,

[18] Henkel, C., Bubeck, A., & Xu, W. (2016). Energy Efficient Dynamic Window Approach for Local Path Planning in Mobile Service Robotics. IFAC-PapersOnLine, 49(15), 32–37. doi: 10.1016/j.ifacol.2016.07.610

[19] Ratsamee, P., Mae, Y., Ohara, K., Takubo, T., & Arai, T. (2012). Modified social force model with face pose for human collision avoidance. 2012 7th ACM/IEEE International Conference on Human-Robot Interaction (HRI), 215–216. doi: 10.1145/2157689.2157762

$F^{facepose}$ is summed with the other forces, resulting in: $\sum F = F^{goal} + F^{object} + F^{human} + F^{facepose}$
A path planning for robot R and a motion prediction for human H is then derived from the differential equation $\frac{d}{dt}\vec{v} = \frac{\sum F}{m}$.

The following figure shows an overview of calculated forces acting on a human (H) and a robot (R) during collision avoidance.

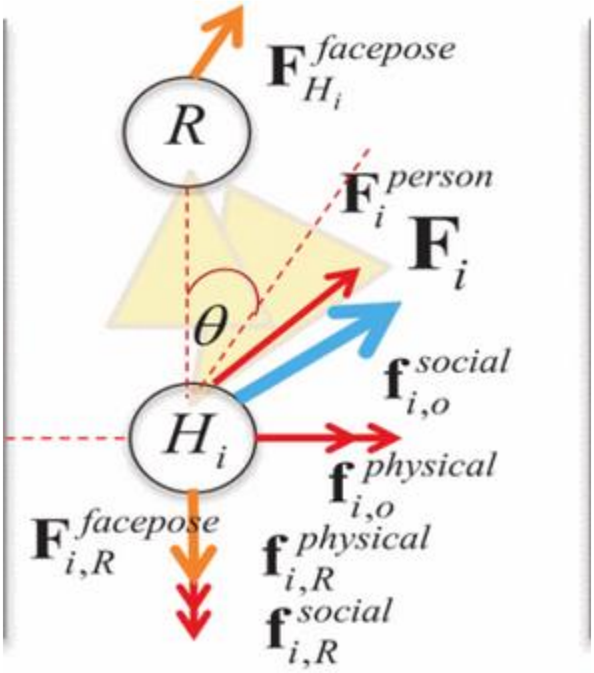

Figure 5 overview of forces from the social force model, by Ratsamee, P. et al.

Now that this approach is briefly described, it is important to look at which user requirements are satisfied and if this model possibly needs adaptations or extensions for use in the supermarket collision avoidance environment described previously.

First of all, this model can nicely incorporate the notion of a human's personal space through f^{social} and $f^{physical}$ that can be defined so that f^{social} grows in size when entering someone's personal space. This roughly satisfies user requirement 1, although its definition greatly influences how well this user requirement is satisfied. For example, these repulsive forces might be too strongly present resulting in the robot never entering someone's personal space which in turn could lead to having no way to avoid crowded spaces. User requirements 2, 4 and 5 can also be satisfied. Because the robot tracks the face pose of nearby humans, it is able to plan a predictable and non-erratic path around a human that also adapts it velocities accordingly. Therefore, the following problems are taken care of by the model:

Blocking a human's path as described under user requirement 2 is evidently avoided because the approach will detect the human beforehand and plan a path around it. This is of course under the assumption that either the ceiling mounted camera or the robot's local camera senses this human.

A discomforting velocity as described under user requirement 4 is prevented, because the velocity is adapted according to the previously described differential equation which considers all social and physical repulsive forces. These forces need to be calibrated well, so that moving closer to human does indeed result in lower robot velocities. There are however extensions needed to constrain these velocities; under every condition a robot's speed should have a maximum value of 1.4 [m/s]. Also, a minimum velocity of 0.5 [m/s] should be chosen when the robot happens to be (briefly) in someone's intimate space and not obstructed by someone's physical space. In that way, user requirement 4 should satisfied.

Erratic motions as described under user requirement 5 are avoided if all the forces calculated do not change significantly in a short time span leading to robot paths changing rapidly. Correct placement of sensors on the robot or the environment can prevent this. To further prevent erratic velocity profiles, a maximum robot acceleration based on the average pedestrian acceleration of 0.68 [m/s^2] should be added.

User requirement 7 might also be partially satisfied as the path planning algorithm makes use of a person's gaze through $F^{facepose}$. It is thought that this makes the path planning very predictable and readable for surrounding humans. This would make the approach more easily excepted for humans in general, probably also in a variety of countries and cultures. Empirical evidence needs to be found that this path planning model for robots is indeed predictable and generally accepted by humans, however, Wang, P. [BRON][20] says that the social force model already is consistent with psychological findings regarding for example (interpersonal) stress.

In conclusion, this approach needs no extension with motion prediction of moving objects, because the algorithm presented works on both humans (for prediction) and robots (for path planning) and combines both to form a robot path. Overall, both humans and objects can be avoided in a desired way in conformity with most user requirements after some adaptations. However, it might be desirable to add cost functions to the static environment like object padding for the aisles and object occlusion (hidden zones). This approach can also benefit from the use of ceiling mounted cameras in the environment, because then blank spots or errors in the local sensing of the robot, possibly causing erratic calculations of forces, can be avoided. Static objects already in the environment can also be detected by these cameras, which is beneficial for this approach when calculating object repulsive forces.

A disadvantage of this approach might arise in the case of peak customer times where some supermarket aisles can be densely crowded. When large groups of people are walking around or standing in an aisle, there is a significant increase in the amount of forces that need to be calculated in real time, which might lead to a decrease in robot reactiveness to the environment. It needs to be investigated how many calculations are admissible to keep robot reactiveness.

Another disadvantage inherent to the social force model lies in the fact that humans and robots are defined as particles with no physical radius. In a supermarket application this simplification cannot be made, because avoiding actual physical collision should of course be considered as a top priority.

A solution might be found when looking at Zeng and Bone's work [BRON][21]. This approach to collision avoidance also makes use of repulsive and attractive forces, just like the social force model. However, it clearly defines critical regions around humans, dynamic objects and static object. This area is defined in the following figure:

[20] Wang, P. (2016). Understanding Social-Force Model in Psychological Principles of Collective Behavior. arXiv, 1605.05146. Retrieved from https://arxiv.org/abs/1605.05146

[21] Zeng, L., & Bone, G. M. (2013). Mobile Robot Collision Avoidance in Human Environments. Int. J. Adv. Rob. Syst., 10(1), 41. doi: 10.5772/54933

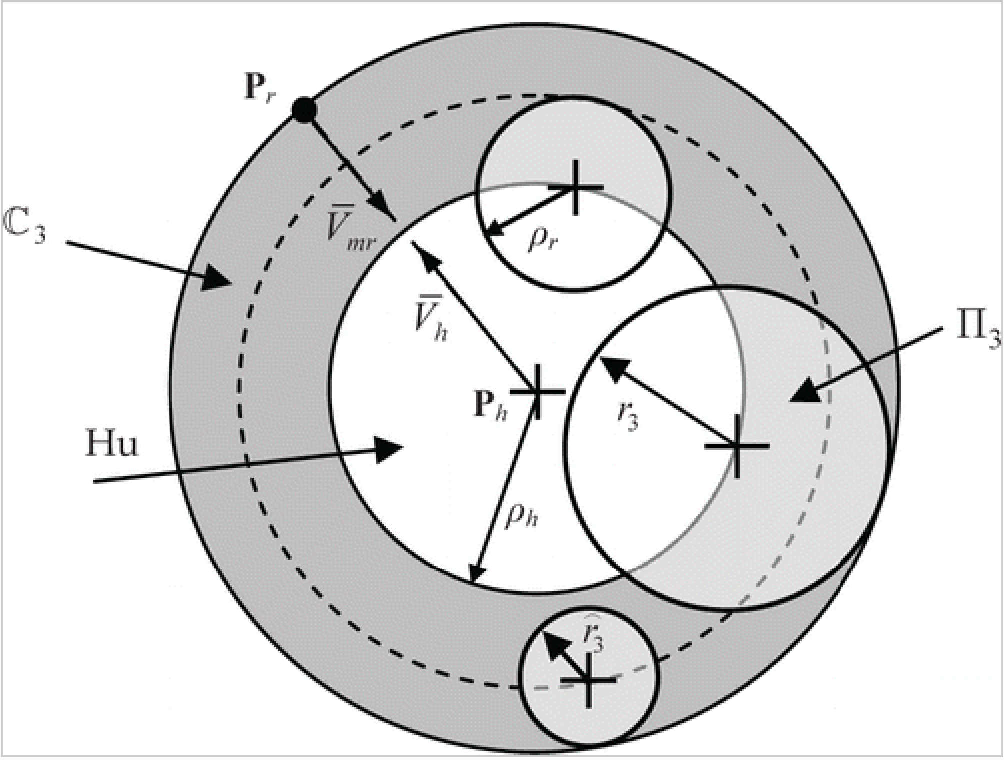

Figure 6 The critical region for a human as defined by Zeng & Bone.

Since the robot will move on the floor, a human should be modelled as the projected shape of its body on the floor. This projected shape is dependent on the pose of the human, but by modelling a human body as a cylinder the different poses are neglected, while still maintaining safety. Since the average step length of a human is 0.8 m [BRON][22], the radius of this cylinder ρ_h is taken as 0.4 m. The robot's radius is defined as ρ_r, so when the robot's centre is located along the dashed line, it makes contact with the human. This approach states that the robot should decelerate with maximum deceleration when it enters C_3. Since this critical region will only be entered by the robot in a worst-case scenario, another region should be defined where collision avoidance can take place. Zeng & Bone define this as follows:

[22] Martin, P., Marsh, A. (1992) Step length and frequency effects on ground reaction forces during walking, Journal of Biomechanics, Vol. 25, No. 10, 1992, pp. 1237–1239

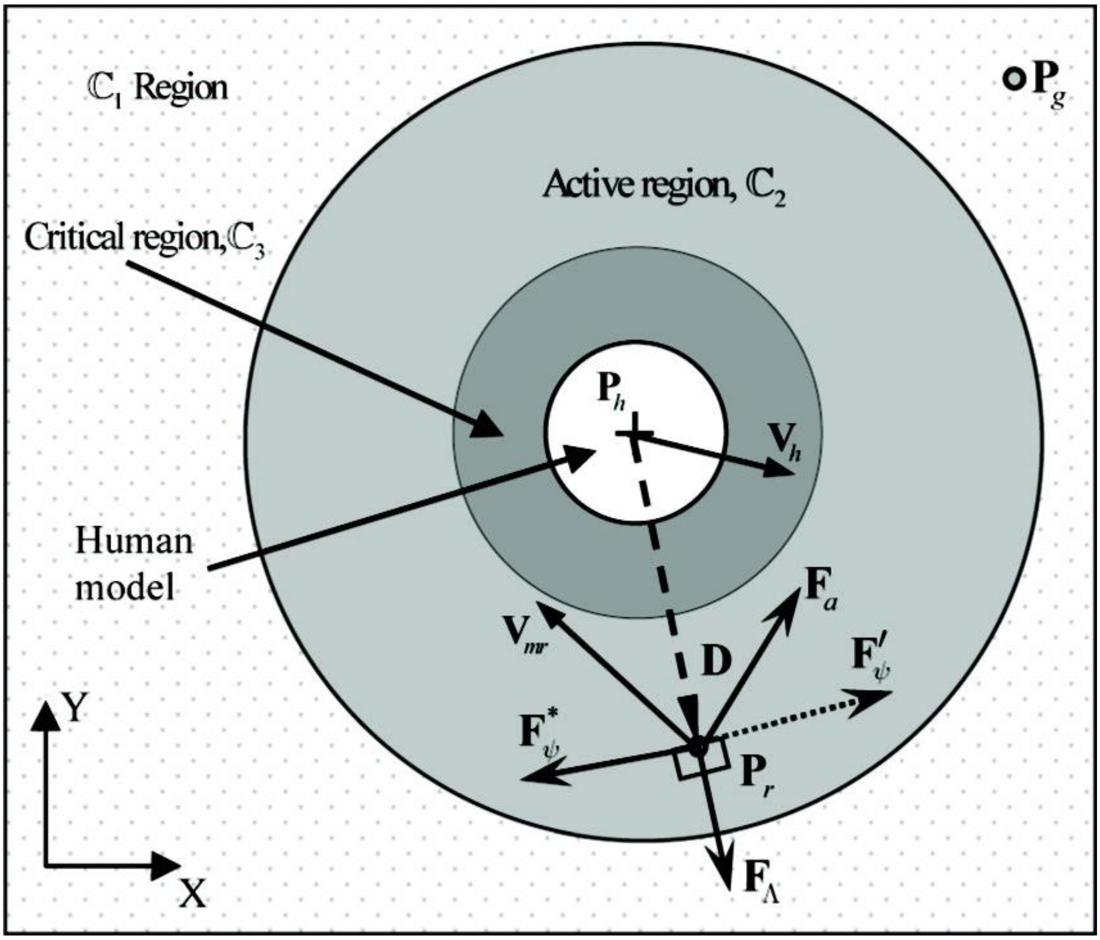

Figure 7 The active region of a human where collision avoidance should take place

This active region is where social repulsive forces should become larger than zero, so the trajectory and velocity of the robot can be adapted for collision avoidance.

Another aspect that the SFM does not consider is that humans sometimes move around in groups, which is especially the case in crowded environments. Because these people generally want to stay inside the group there should be extra attractive forces added so that group cohesion is maintained as much as possible. It was found that the Headed Social Force model introduced by Farida, F et al [BRON][23] incorporates this aspect.

It can be concluded that the most viable approach is given by the Social Force model, mainly because the Dynamic Window approach only takes into account static objects, while the environment of a supermarket consists of many moving objects. A distinction between inanimate objects and humans is also necessary, which the DWA does not consider. Significant adaptations need to be made to make this approach viable here, so it is considered inferior. The social force model on the other hand can relatively easy be extended with extra forces or environment cost functions and constraints to better fit the user requirements for a supermarket.

[23] Farina, F., Fontanelli, D., Garulli, A., Giannitrapani, A., & Prattichizzo, D. (2017). Walking Ahead: The Headed Social Force Model. PLoS One, 12(1), e0169734. doi: 10.1371/journal.pone.0169734

Now that the Social Force model is considered as the most viable collision avoidance option and necessary extensions are described, a simulation approach to test the viability of this adapted approach is elaborated on.

Candidate procedure simulation

The main aim of the simulation is to show that the adapted social force model is indeed suitable for application in a supermarket environment, while the standard social force model is not. The concept of a collision avoidance environment in the previous section will be made explicit by giving its exact geometries, describing all agents and obstacles present in the environment. Furthermore, formulae will be presented by which this algorithm can be tested in how well certain user requirements are satisfied.

The headed social force model (HSFM) by Farina, F. Et al will be used as a basis for this simulation, since a MATLAB implementation is readily available, and it already incorporates the modelling of humans as cylinders instead of particles. Furthermore, it adds the concept of a human's heading which is in this case used to describe the direction in which humans tend to walk. This is necessary because it has been empirically shown [BRON][24] that people prefer to walk forward most of the time, while lateral displacements are rarely seen. This model also adds forces so that human group cohesion is maintained as much as possible. Unless stated otherwise, the model parameters from Farina, F et al. will be used.

Environment simulation with HSFM

The simulation environment is defined as follows:

- A corridor with a width of 3 [m], which is slightly above the average width of a supermarket aisle according to Steenblock, S., A. [BRON][25]
- The length of the corridor is taken sufficiently large: 10 [m]
- Four small obstacles of variable dimensions to account for fallen retail product, located near the corridor walls.
- Three parked shopping carts, with approximate dimensions W x L of 0.60 x 1 [m], which are the dimensions of shopping carts sold by shoppingcartmart.com [BRON][26]. They can be arbitrarily placed near the walls, so the shopping carts will all be placed near the left wall. The geometry with dimensions given in inches is seen in following figure and is also used in the simulation:

[24] Arechavaleta G, Laumond JP, Hicheur H, Berthoz A. On the nonholonomic nature of human locomotion. Autonomous Robots. 2008;25(1–2):25–35. doi: 10.1007/s10514-007-9075-2
[25] Shelley A. Steenblock. (2010). User centred design evaluation of the grocery store environment. Graduate Theses and Dissertations. 11345. doi: 10.31274/etd-180810-2791
[26] Supermarket Shopping Cart. (2018, October 10). Retrieved from https://www.shoppingcartmart.com/supermarket-shopping-cart.html

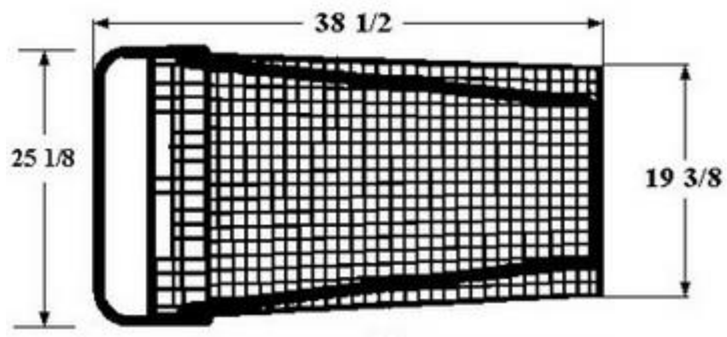

Figure 8 Dimensions in inches of a shopping cart

Implementation in the HSFM by Farida, F. et al gives the following environment:

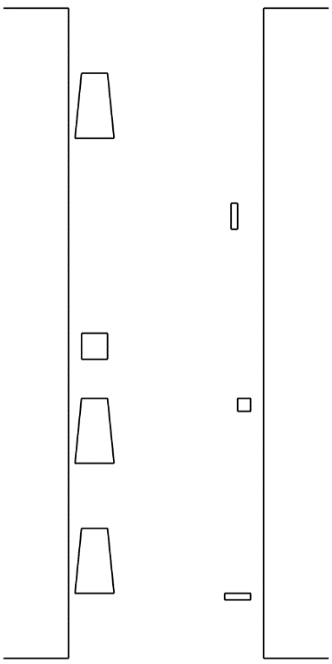

Figure 9 The simulation environment of a supermarket aisle in the HSFM

The agents present in the environment and their goals are described now.

All humans in the simulation will have the following characteristics: their radius ρ_h varies between 0.35 and 0.45 [m], their mass varies between 60 and 90 [kg] and their desired speed will vary between 0.5 and 1.4 [m/s].

In this model, the robot could have about the same characteristics as humans, because changing the characteristics of the robot does not have a great impact on the simulation. However, the robot can have a significantly smaller radius than humans, so taking ρ_r = 0.2 [m] should suffice.

The agent groups that should be added to the environment:

- Two groups, each consisting of 3 humans, each group entering the corridor from opposing direction will walk through the environment. Their goal thus lies outside of the corridor.
- Two groups of humans, both consisting of 2 humans will enter from either side having a distinct goal present at the walls of the corridor and will then move outside the corridor.
- Two humans will be stationary somewhere in the corridor.

- One robot will enter the corridor from the bottom and has its goal at the top end, to simulate a robot navigating through.

A (technical) problem with this model is that it makes use of groups of people that need to have at least two humans in them. Because of this, the robot needs to be programmed in separately. It was found that the first three agent groups can be added to the model without much adaptation.

Simulating only the first two agent groups gave the following result:

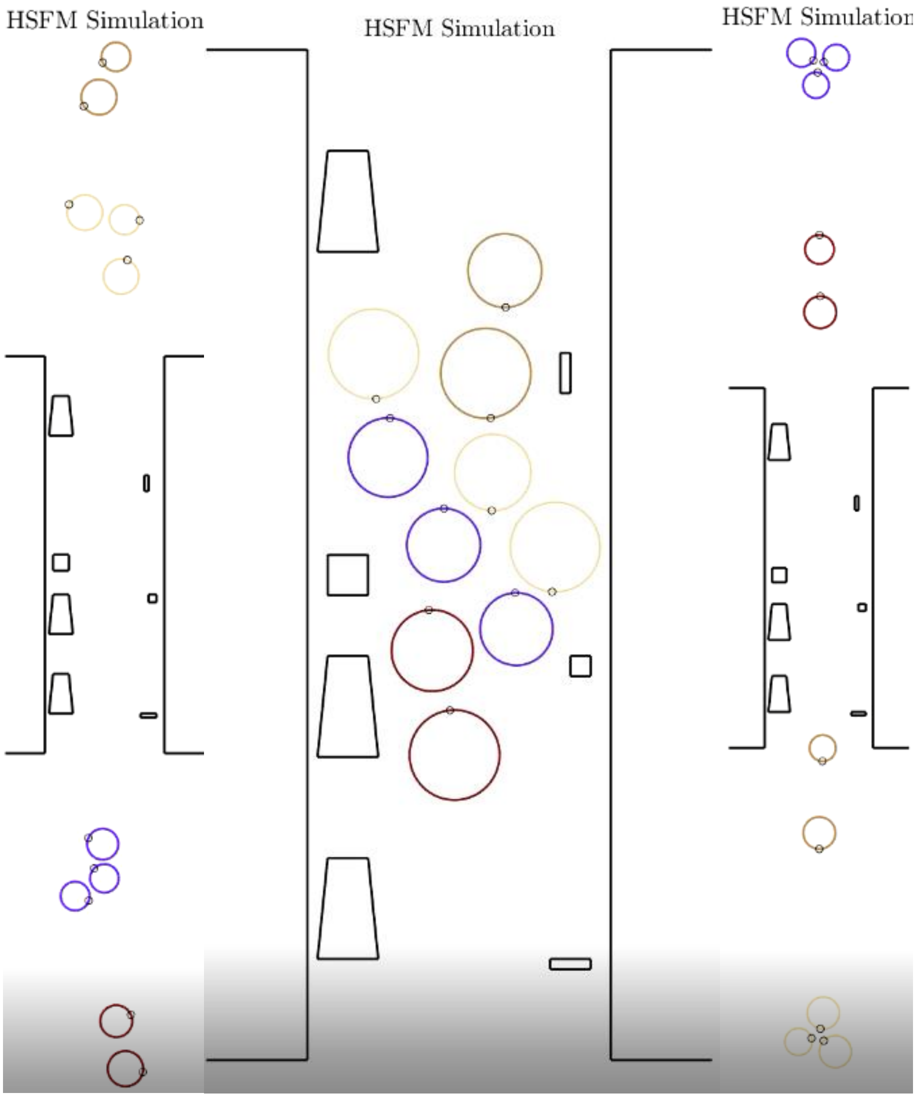

Figure 10 from left to right: Initial condition - humans avoiding each other - final condition (40 time steps)

Adding the third agent group leads to the following two final conditions:

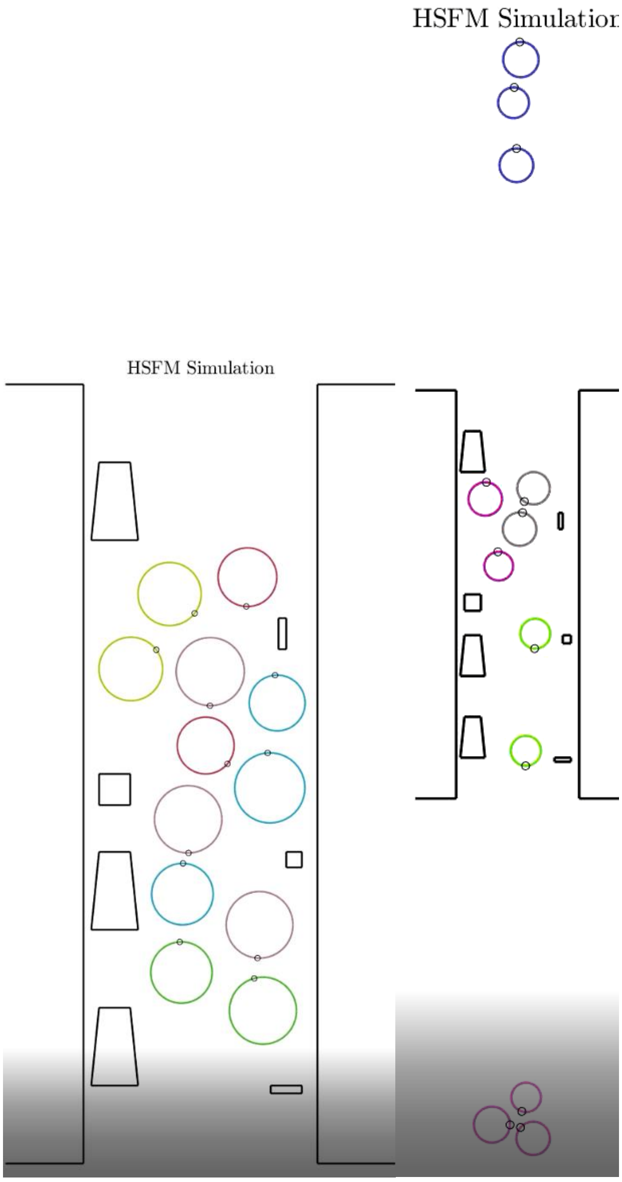

Figure 11 the final conditions after 40 time steps, simulating with the first three agent groups

A video of these simulations is also available on the wiki.

Testing user requirements

After the robot has been implemented in the model, the model should track how many problems with regards to HRIs (and thus user requirements) have occurred.

By calculating the amount of erroneous actions taken by the robot in this environment and relating this to a negative score, a conclusion can be drawn on whether user requirements are satisfied better with the adapted approach compared to the standard SFM.

Therefore, the simulation needs to keep track of the following:

- How many times actual physical collisions happen and at what velocity (related to user requirement 1).

Since this is the most undesired, this action will be highly weighted in calculations. Higher collision velocity should add more negative points. This can be implemented as follows (in pseudo-code):

A physical collision occurs:

If: the distance from the centre of the robot to the centre of the nearest human equals $\rho_h + \rho_r$. (This distance can never be less than $\rho_h + \rho_r$)

Then: the robot's velocity relative to the human at this time ($v_{r,h,i}$) needs to be stored.

The total negative points P_c for the robot due to physical collision can then be calculated with:

$$P_c = w_c * \sum_{i=0}^{n} v_{r,h,i}$$

Where,

w_c is a weight factor for this calculation, which should be substantially higher than the other negative points calculations. Taking the value for w_c 10 times higher than for other calculations should implement the undesirability of physical collision happening.

$v_{r,h,i}$ is the relative robot velocity at physical collision i

n is the total number of physical collisions between human and robot that happened during simulation.

- For which amount of time and to what degree a robot has entered someone's personal space and with what velocity. (related to user requirement 1 and 4)

The degree in which this is undesired for humans is dependent on what velocity the robot used while moving in a human's personal space and how far into someone's personal space a robot moved in. A high velocity results in more negative points P_p.

A human's personal space is quite large, having a radius of 1.2 [m] around a human. The cost function approach by Barnaud, M-L. Et al which was already described can be used here. The personal space cost will then be modelled as two 2D normal distributions joined together seamlessly. The normal distributions have an independent front variance σ_h and rear variance σ_r but have the same side variance σ_s. Their research showed that values for σ_h, σ_r and σ_s of 1.5, 1.3 and 1.5 (respectively) showed the best fit with experimental results. This means that this normal distribution is slightly elongated to a human's heading.

So, if a robot enters someone's personal space from behind the cost function is defined as:

$$\mathcal{N}(x, y \; ; \; [x_0, y_0], \Sigma_r)$$

If the robot enters someone's personal space from the front the cost function is:

$$\mathcal{N}(x, y \; ; \; [x_0, y_0], \Sigma_h)$$

Where x_0 and y_0 are the coordinates of the centre of a human, thus the centre of the 2D normal distribution, x and y the local coordinates for a human and with covariance matrices:

$$\Sigma_h = \begin{pmatrix} \sigma_h & 0 \\ 0 & \sigma_s \end{pmatrix}, \; \Sigma_r = \begin{pmatrix} \sigma_r & 0 \\ 0 & \sigma_s \end{pmatrix}$$

Calculations with this model will be stored in PS and are initially set to 0.

Every time a robot enters someone's personal space the distance between the centre of the robot and the centre of the human is less than $\rho_h + \rho_r + 1.2$

While this holds:

> PS(x,y) should be evaluated and multiplied by the robot's relative velocity to the human for every time step. This value then needs to be added to PS.

We can then derive the following formula for calculating the total negative points P for entering a human's personal space during the entire simulation:

$$P_p = w_p * \sum_{i=0}^{n} v_{r,h,i} * PS_i(x,y)$$

$PS_i(x,y)$ is the evaluation of the 2D normal distribution at time step i and local coordinates x and y of the human. n is the total number of time steps the robot was present in someone's personal space.

w_p is the weight for this calculation

- Acceleration values of the robot (related to user requirement 5)

User requirement 5 simply states that the acceleration of a robot may not be higher than 0.68 [m/s^2]

This can be checked by storing the robot's acceleration and calculate the difference between that acceleration and 0.68 for every time step i. Every non-negative value (a_i) will then be summed for the entire simulation to calculate the negative points due to acceleration P_a:

$$P_a = w_a * \sum_{i=0}^{n} a_i$$

w_a is the weight for this calculation

n is then the total number of time steps in the simulation

The total negative points for the entire simulation is then:

$$P = P_c + P_p + P_a$$

User requirements 1, 4 and 5 can now be quantitatively assessed for the standard and adapted SFM in a simulation by comparing their P-values.

HSFM extensions

The first extension that needs implementation is adding $F^{facepose}$ to the HSFM. In the HSFM, θ_i is the heading, which is defined as the angle between x-axis of the body frame centred at the human's position and the x-axis of the global environment.

Then, the distance between human and robot $d_{i,r}$, the sum of radii of human and a robot $r_{i,R}$ and this θ_i need to be used to calculate $F^{facepose}$, according to the formula of $F^{facepose}$:

$$F^{facepose} = FS * e^{\frac{r_{i,R} - d_{i,R}}{s_R}} * \overrightarrow{\boldsymbol{v_{i,R}}} * (\lambda + (1 + \lambda)\frac{1+cos(\theta)}{2}$$

Where θ is now defined as the difference in heading of a human $\theta_{i,H}$ and a robot $\theta_{i,R}$:

$$\theta = \theta_{i,H} - \theta_{i,R}$$

In the HSFM, f_i is denoted as the total force acting on an individual, $f_{i,0}$ as the attractive force to the goal and $f_{i,e}$ describes the repulsive and interaction forces. $F^{facepose}$ should thus be added to $f_{i,e}$:

$$f_i = f_{i,0} + f_{i,e}$$

And with:

$$f_{i,e} = f_{i,p} + f_{i,w} + F^{facepose}$$

The next extension is adding (static) cost functions to the environment. It is thought that the object occlusion, hidden zone and object padding cost functions are only necessary to be implemented in the HSFM. The other cost functions proposed previously would not be necessary to add, because they are already described well through the HSFM. Small static obstacles on the ground are already taken care of by the HSFM, but for the walls (supermarket shelves) cost functions need to be added.

These cost functions should have a direct impact on velocities and should thus skip the force calculations.

For object padding a cost C_p is defined, which is inversely proportional to the robot's distance to the walls $d_{i,w}$:

Only if the $d_{i,w}$ is smaller than or equal to 2 times the robot radius ρ_r, this cost function should be evaluated so that movement further away from the walls is not affected:

$$C_p = \frac{1}{d_{i,w}} \text{ (if } d_{i,w} \leq 2\ \rho_r\text{)}$$

For object occlusion and hidden zones, a cost C_o is defined which is now inversely proportional to the distance to the centre of a certain critical zone $r_{i,c}$. In this environment critical zones are the ends of the corridor, where the robot might not expect people coming around corners.

The distance to the centre of this critical zone is defined as $d_{i,c}$. 2 ρ_r is now taken as the radius of this critical zone, leading to:

$$C_o = \frac{1}{d_{i,c}} \text{ (if } d_{i,c} \leq 2\ \rho_r\text{)}$$

Where

$$d_{i,c} = \left\| r_{i,R} - r_{i,c} \right\|$$

$r_{i,R}$ represent the coordinates of the robot.

The coordinates of this critical zone's centre ($r_{i,c}$) should of course be programmed in manually.

The two cost functions can then be summed, applying weight factors:

$$C = w_{padding} * C_p + w_{occlusion} * C_o$$

The robot velocity $v_{i,R}$ should then be reduced, according to:

$$v_{i,R} = v_{i,R} - C$$

To make sure this does not stop the robot completely, the minimum velocity of 0.5 [m/s] should always be chosen if C_p gets too high.

A visualisation of the proposed cost functions to the simulation environment is given in the following figure:

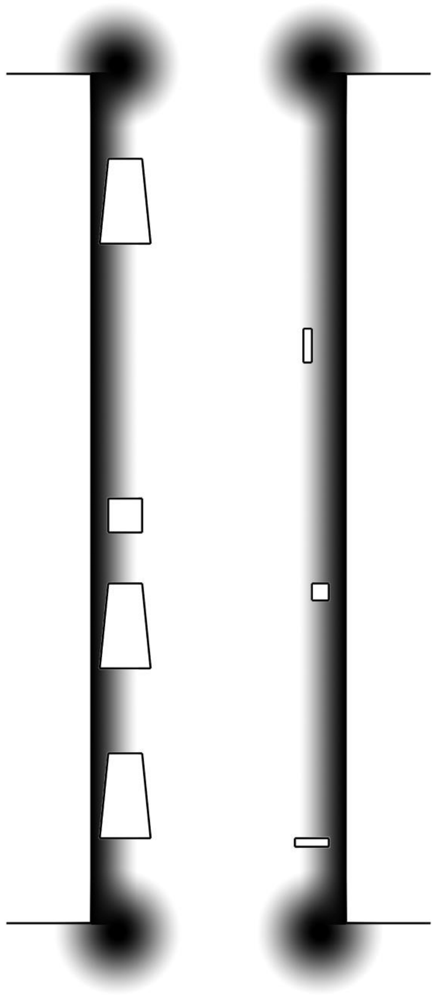

Figure 12 cost function mapping to the simulation environment

A simulation that implements the user requirement tests, with the extended HSFM should prove that this approach is superior for collision avoidance in a supermarket.

Discussion

Environment simulation with HSFM

Looking at the simulation with the standard SFM model (figure 10), it can be seen that there is already quite a complex situation for robot collision avoidance when only 10 humans are moving in and out of the corridor. It should also be noted that the environment is not very ‘strict’; a wider than average corridor is chosen, and the parked shopping carts are all relatively out of the way. If a robot would be introduced with smaller radius but equal characteristics as humans, it can probably already lead to the robot getting stuck and some user requirements might be violated. However, all groups do reach their destination after 40 time steps.

Then adding two stationary humans leads to the model having a hard time to get all groups to their destinations in 40 time steps. Sometimes no destinations were reached at all, or only 2 out of 5 groups reached their destination. Also, computation times increase with a factor of 6.

Cost functions

This minimum velocity used in cost functions is the admissible approaching speed for robots interacting with humans described earlier, so this might need (slight) adaptation.

A proper weighting in cost functions needs to be found.

Extended HSFM

The extended HSFM is thought to be superior than the regular SFM for robot collision avoidance, because of the following points:

Topics for further research

How many extra ceiling cameras would be necessary and how much would that cost?

However, it needs to be investigated whether it is still desirable for a supermarket enterprise to invest in more cameras, when the cheaper option of only using local robot cameras might suffice.

It should be investigated what kind of cues are desirable in crowded situations; it will still be necessary to do further research on which exact sentences using low controlling language are the most effective and fitting in for this context, minimising social reactance.

Furthermore, it needs to be looked at how (computationally) inefficient some collision avoidance procedures become when large groups of people are in the robot's vicinity and how these inefficiencies can be overcome.

It needs to be investigated if this motion prediction for shopping carts is worth the extra computational cost to the algorithm, possibly by doing simulations in combination with real-life experiments. It needs to be investigated how many calculations are admissible to keep robot reactiveness

Empirical evidence needs to be found that this SFM model for robots is indeed predictable and generally accepted by humans

Calibration and validation of the SFM is necessary according to Tang, M et al [BRON][27], so also the adapted SFM presented here will need that.

Conclusion

This research described the difficulties and advantages for robot collision avoidance in a supermarket environment and then formulated design requirements based on the users in this environment, being customers and staff members. A basic idea was presented that these user requirements can be incorporated in cost functions. Then, collision avoidance through the Dynamic Window Approach and the Social Force Model were assessed on how well they would satisfy user requirements and how fitting they were for the supermarket environment. The Social Force Model was chosen as the most viable option. This was then followed by a description on how a simulation can be done that quantitatively tests the proposed user requirements for the SFM. Moreover, a proposed extension to the Headed Social Force Model was described that is thought to be the superior option for collision avoidance in a supermarket. The research unfortunately failed to provide the technical

[27] Tang, M., & Jia, H. (2011). An approach for calibration and validation of the social force pedestrian model. Proceedings 2011 International Conference on Transportation, Mechanical, and Electrical Engineering (TMEE), 2026–2031. doi: 10.1109/TMEE.2011.6199614

implementation of the user requirements tests and the adapted SFM, however it did describe in a more conceptual way how implementations can be achieved.