Depth Sensor Placement for Human Robot Cooperation

Max Stähr, Andrew M. Wallace and Neil Robertson
School of Engineering and Physical Sciences, Heriot Watt University, Edinburgh, U.K.

Keywords: Mobile Robot Sensor Placement, Collision Avoidance, Human Robot Cooperation.

Abstract: Continuous sensing of the environment from a mobile robot perspective can prevent harmful collisions between human and mobile service robots. However, the overall collision avoidance performance depends strongly on the optimal placement of multiple depth sensors on the mobile robot and maintains flexibility of the working area. In this paper, we present a novel approach to optimal sensor placement based on the visibility of the human in the robot environment combined with a quantified risk of collision. Human visibility is determined by ray tracing from all possible camera positions on the robot surface, quantifying safety based on the speed and direction of the robot throughout a pre-determined task. A cost function based on discrete cells is formulated and solved numerically for two scenarios of increasing complexity, using a CUDA implementation to reduce computation time.

1 INTRODUCTION

Mobile service robots have significant potential for factory automation, however safety is of paramount importance. The main concern in human robot cooperation is sudden, unforeseen contact between a robot and a human and of severe force and pressures on the human body. Approaches to safe human robot cooperation can be divided into two categories: limiting the consequences of an impact and avoiding a collision. In the former case, the injuries of unexpected human-robot impacts can be limited by lightweight/compliant mechanical design of the manipulator and post-collision reaction strategies (De Luca et al., 2006).

However, reducing risk by mechanical design also reduces the payload of the robot system, and safety may well be compromised in unstructured environments. In the latter case, collision avoidance based on exteroceptive sensors is independent of the payload and mass of the robot (Angerer et al., 2012). Early developments utilized fixed, mounted intensity cameras (Ebert and Henrich, 2002) (Henrich and Gecks, 2008) to detect overlapping silhouettes of robots and obstacles. Later developments used 3D environment perception in real time to improve collision avoidance for human robot cooperation and Euclidean distance related robot velocity controls (Fischer and Henrich, 2009) (Graf et al., 2010) (Lenz, 2012). However, surveillance of the operational area of a mobile robot by a static sensor network is undesirable because many sensors are required to cover a large area, and it lacks flexibility if the task or operating environment is changed. Rather, mounting the sensors on the robot so that they have a dynamic perspective gives much greater flexibility.

The major drawback of this approach is the probable interaction between the robot-mounted sensors and the mobile platform and manipulators which can result in large shadowed volumes and consequent unseen collisions. A necessary strategy is to raise the number of sensors to decrease shadowed volumes, but computational and power resources are limited. The problem of optimal placement becomes even more challenging as the sensor positions change with robot movement.

1.1 Related Work

We pose the problem as one of optimization with respect to a cost function that maximises the visibility with regard to regions of potential robot human collision. Systematic optimization of static camera setups is a known challenge in surveillance tasks, aiming to minimize the demanded resources while concurrently maximizing surveillance. The visibility aspect of the problem can be regarded as an extension of the art gallery problem which consists of finding the minimum number of cameras to survey a given space (O'rourke, 1987). The distinction from the orig-

Stähr M., M. Wallace A. and M. Robertson N..
Depth Sensor Placement for Human Robot Cooperation.
DOI: 10.5220/0005017103110318
In *Proceedings of the 11th International Conference on Informatics in Control, Automation and Robotics* (ICINCO-2014), pages 311-318
ISBN: 978-989-758-040-6

inal art gallery problem is that our guard is moving along on a pre-defined robot path and our objective is to maximise visibility of regions where the possibility of collision is greatest.

Numerical solutions decompose the workspace into equal sized cells and assign to each a binary attribute indicating whether or not it is seen by a sensor at a certain position. The number of sensors is raised and their positions are varied until the optimum is found (Dhillon et al., 2002). Often the system performance has been described as the detection of certain features in the surveyed scene such as humans (Yao et al., 2008), (Bodor et al., 2007) or robot paths (Nikolaidis et al., 2009) which are not equally distributed over the grid. A cost function is utilized to factor the quality of feature recognition at a certain cell from a known sensor position. Including the presence of occluders, e.g. other humans (Mittal and Davis, 2008) or pillars (Chen and Davis, 2008) improves the overall performance of such systems in dynamic surroundings.

The grid is extended to a voxel grid for a three dimensional feature distribution. In the work of Flacco (Flacco and De Luca, 2010) the position of depth and presence sensors is optimized for a fixed robot arm and human sharing the same workspace. A probabilistic visibility approach for pre-defined robot motion and human presence distribution is delineated. Sensor mounting positions are restricted to the surrounding wall and obstructions include the robot arm and the human itself as well as static environment objects such as walls and tables. We extend the approach of Flacco to dynamic environments with pre-defined human and robot movements. The major extensions to Flacco's work presented here are the use of moving cameras on the mobile robot and the introduction of a safety criterion which reflects the consequences of human robot collisions.

Concepts of robot safety were developed mainly in human aware robot motion planners. The human position is determined and the robot's velocity or trajectory is adapted according to the distance (Haddadin, 2013), robot arms inertia (Kulic and Croft, 2007) and direction (Sisbot et al., 2010). More complex approaches build up velocity related scalar fields for each robot arm link and superimpose them on a resulting field according to the current joint configuration and speed (Lacevic and Rocco, 2010). The number of sample points is bounded to fulfil real time constraints of the motion controllers. Here, we extend the scalar field of Lacevic to the robot mesh and apply it to arbitrary and complex robot shapes and geometric structures because the source of danger is not the geometric centre of link, rather the solid outer shape of the robot.

We have chosen to formulate a numeric approach for optimal depth sensor placement for collision avoidance on mobile robots. The working area is decomposed into cells of equal size and each cell is assigned with both binary visibility and scalar danger attributes representing the potential collision in this cell. We specify the vertices of a mesh of the outer robot shape as potential sensor positions and sample all three rotational degrees of freedom for each position. A cost function depending on the sensor's position and orientation is introduced and numerically minimized by complete enumeration as this is an offline process.

This paper is organized as follows. Sections 2 and 3 describe the robot, human and sensor model as well as the work cell decomposition based on visibility. Section 4 introduces the quantifying safety measurement utilized in the cost function introduced in section 5. Section 6 provides two experiments with increasing complexity and section 7 summarises the results and gives an outlook on future work.

2 ROBOT AND HUMAN MODELING

We model the robot as an articulated, rigid body chain consisting of static meshes and time variant homogeneous transformations T_t at time t. The position and orientation of the *m*-th link w. r. t. a global coordinate origin can be expressed by:

$$^0T_m = {}^0T_1 \dots {}^{m-1}T_m \tag{1}$$

in which $^{i-1}T_i$ describes the homogenous link-to-link transformation following the Denavit-Hartenberg-convention and mapping a robot state vector $q_{r,t}$ as

$$q_{r,t} = [x_{r,t}\ y_{r,t}\ \theta_{r,t}\ a_{1,t} \cdots a_{6,t}]^T \tag{2}$$

in which $x_{r,t}$, $y_{r,t}$, $\theta_{r,t}$ expresses the translational and rotational displacement of the mobile platform associated with the mobile robot frame $T_{mr,t}$ and $a_{1,t} \cdots a_{6,t}$ are the joint angles of the 6R (6 rotatory joints) robot arm at time t. The complete robot shape RM_t is defined as the union of all meshes M_n transformed according to their parent link frame $T_{n,t}$ at time t.

The human is represented as a rigid stick figure considering Cartesian position and gaze direction. Following that simplification the human state vector $q_{h,t}$ at time t is expressed as:

$$q_{h,t} = [x_{h,t}\ y_{h,t}\ \theta_{h,t}]^T \tag{3}$$

and associated with the frame $T_{h,t}$ having an equivalent state vector to the mobile platform.

3 SENSOR MODELING

Each of the depth sensors is modelled as a virtual camera. Each ray that passes through the focal centre F_c assigns its data to a point on the image plane IP. The image coordinates x on IP are expressed by the pin hole projection equation:

$$x = PX = KEX \tag{4}$$

The intrinsic parameters are stored in the camera matrix K, and the extrinsic homogenous transformation E is used to obtain the projection matrix P. All Cartesian points X on the same ray lead to the same value of x. The depth of a Cartesian point to a known camera position is given by:

$$depth(X) = \|F_c - X\| \tag{5}$$

We store the depth information along the ray of the corresponding pixel on the IP and name it depending on its source.

3.1 Sensor Position

As we mount the sensor on the mobile robot the position and orientation of the sensor depends at least on the frame $T_{mr,t}$, thus our objective is to optimize the sensor position vector q_c w.r.t to $T_{mr,t}$:

$$q_c = [x\, y\, z\, roll\ pitch\, yaw]^T \tag{6}$$

In this work we make a simplification and attach the focal centre of the camera to one point of the complete initial robot mesh RM_i for $a_1 \dots a_6 = 0$. On a real robot the focal centre cannot be attached directly to the robot hull as there are mounting constraints, such as minimum distance to the hull or forbidden regions such as drives, gripper, wheels or laser scanners. In this work we define the set of all possible camera position on the robot as:

$$\begin{aligned} P_c = \{q_c \in \mathbb{R}^6 : & q_{c,1} \dots q_{c,3} \in RM_i \wedge \\ & -\pi < q_{c,4} \dots q_{c,6} \leq \pi\} \end{aligned} \tag{7}$$

As the camera may be attached to the robot arm, its position and orientation may vary for any $a_1 \dots a_6 \neq 0$. We compute the resulting position and orientation of the sensor by determining the local, time invariant transformation ${}^{T_n}T_c$ between the camera frame T_c in the base of the link frame T_n to which it is attached, building the closed kinematic chain via T_{mr}:

$${}^{T_m}T_c = {}^{mr}T_{T_m}{}^{-1}\, {}^{mr}T_c \ \forall\ q_r(a_1 \dots a_6 = 0) \tag{8}$$

Now we can extend the kinematic chain from equation 1 by the time invariant transformation ${}^{T_m}T_c$ to:

$${}^{mr}T_c = {}^{0}T_1 \dots {}^{m-1}T_m\, {}^{T_m}T_c \ \forall\ q_r \tag{9}$$

and determine the camera frame w.r.t to the mobile platform ${}^{mr}T_c$ for all $q_{r,t}$.

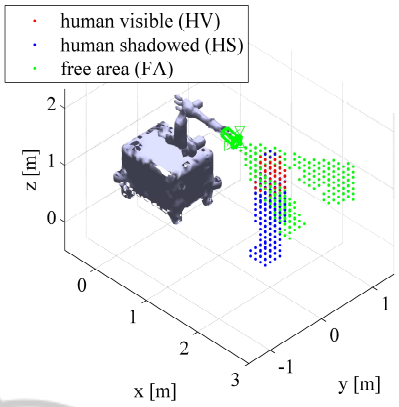

Figure 1: workspace decomposition.

3.2 Work Cell Decomposition

We decompose the mobile platform centered cell WC around a mobile robot in N regular cells of equal edge length. Each cell belongs to one of the following set. The free area FA represents the set of all Cartesian cells which are in the field of view (FOV) of the depth sensor but do not belong to either the robot R, the environment E (static objects) or the human H. The dark area DA sums all the cells outside the FOV for which we have no information. In contrast to previous work Flacco (Flacco and De Luca, 2010) we do not distinguish between sources of DA, rather we maximise the visible human cells HV, which are the difference set of human H and human shadowed HS sets. Summarizing, at any time t, the following relation holds:

$$\begin{aligned} WC(t) &= FA(t) \cup DA(t) \cup R(t) \cup E(t) \cup H(t) \\ H(t) &= HV(t) \cup HS(t) \\ HS(t) &= H(t) \setminus HV(t) \end{aligned} \tag{10}$$

Following the notation of a depth sensor from equation 5, we define two depth images: the environment depth map E_{DM} and the human depth map H_{DM}. For a pixel x associated with Cartesian points lying on the ray $PX = x$ that intercept the robot or the environment, we set

$$E_{DM}(x) = \min_{\substack{X \in R \cup E \\ PX = x}} depth(X) \tag{11}$$

and

$$H_{DM}(x) = \min_{\substack{X \in H \\ PX = x}} depth(X) \tag{12}$$

if it intercepts the human body. As we ray trace the environment and the human separately we can now compare the depth image pixel by pixel to determine

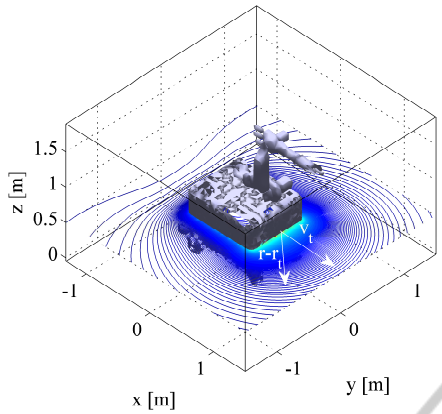

Figure 2: kinetostatic danger field applied to mobile robot at z=0.5m for $\dot{q}_{r,t} = [1.0, 0 \ldots 0]^T$.

the human shadowed volume and free area as follows:

$$FA = \{X \in WC : d_{min} \leq depth(X) < d_{max} \; AND \; H_{DM}(PX) > depth(X) \; AND \; E_{DM}(PX) > depth(X)\} \quad (13)$$

$$HV = \{X \in H : d_{min} \leq depth(X) < d_{max} \; AND \; H_{DM}(PX) \leq depth(X) \; AND \; H_{DM}(PX) < E_{DM}(PX)\} \quad (14)$$

for all Cartesian points. As our goal is to optimize for multiple sensors we define a sensor state vector s for n sensors as the n-times product set of P_c:

$$s \in (P_c)^n \quad (15)$$

representing position and orientation of the nth sensor following the definition of P_c in equation 7. The resulting free and human shadowed areas for n sensors can be described respectively as the union intersections of the corresponding sets for each single camera:

$$FA_s = \bigcap_{c=1\ldots n} FA_c \quad (16)$$

$$HS_s = \bigcup_{c=1\ldots n} HS_c \quad (17)$$

4 SAFETY COSTS

In this work a kinetic-static danger field (KSDF) captures the complete state for safe motion planning of an articulated robot (Lacevic and Rocco, 2010) to quantify safety. Two scalar distance related fields: a quadratic decreasing dynamic field and a linear decreasing static field, are superimposed. The dynamic field does not just consider the distance between the source of danger r_t and sample point r, it also includes the angle between the translational velocity vector $\vec{v}_t$ and the direction vector $r - r_t$ as expressed in equation 18 and illustrated in figure 2.

$$DF(r, r_t, \vec{v}_t) = \frac{k_2 |\vec{v}_t| [\gamma + \cos \angle (r - r_t, \vec{v}_t)]}{|r - r_t|^2} + \frac{k_1}{|r - r_t|} \quad (18)$$

The constant $\gamma \geq 1$ eliminates the change of sign attribute of the cosine. The original field of application of the KSDF is human aware robot motion planning, thus the number of points r on the robot are limited to discrete points on the link of the robot arm between two joints. This approach has been extended to all points of an arbitrary complex robot mesh RM_t at time t as the source of danger is not the geometric link but the solid outer hull of the robot. The meshes have been re-sampled at a unique resolution of 5 cm so the danger field is equally distributed over the robot. The velocity vector v_t is calculated using forward kinematics from equation 1 and the Jacobian of each transformation ${}^{i-1}T_i$.

The danger field is accumulated over all vertices of the robot mesh RM_t for each centre of cell C_i as formulated in equation 19. The resulting scalar field is illustrated in figure 2 for a height of z = 0.5 m and a translational velocity of 1 m/s in the positive x direction.

$$KSDF(C_i) = \sum_{j=0}^{|RM_t|} DF(C_i, RM_t(j), v_t(RM_t(j))) \quad (19)$$

5 OPTIMAL DEPTH SENSOR PLACEMENT

Using the components defined in the previous section, we can compute a cost function with additional assigned weights, w_{HS} and w_{FA} to compute a single scalar value for a given camera state vector s.

$$J(s) = \frac{1}{|P_{rh}| N} \sum_{j=1}^{|P_{rh}|} \sum_{i=1}^{N} w_{HS} HS_{s,j}(C_i) KSDF(C_i) - w_{FA} FA_{s,j}(C_i) KSDF(C_i) \quad (20)$$

for

$$\frac{w_{HS}}{w_{FA}} > \sum_{i=1}^{N} KSDF(C_i)$$

The cost function $J(s)$ sums all the human shadowed cells multiplied by the corresponding KSDF value.

Positions, in which the human is close to the robot path and not seen are prioritised. The cost function is lowered by the weighted cells of *FA* to maximize the visible area for equal human visibility. w_{HS} is much larger than w_{FA} so that maximizing the free area is a secondary objective.

The algorithm for optimal depth sensor placement is divided into two phases: initiation and optimization. During the initialization phase the human shadowed volume and the free area for each possible geometric arrangement are determined as illustrated in algorithm 1. A geometric arrangement is defined as a combination of q_c, $q_{r,t}$ and $q_{h,t}$. We specify the set P_c as the set of all possible q_c with an equivalent P_r for robot positions and P_h for human positions. For each member of P_c we determine *HS* and *FA* for all members of $P_{rh} = P_r \times P_h$. If for at least one member of P_{rh} the relation $H \neq HS$ holds, then q_c is stored in the set of valid camera position P_{valid}, thus the human is visible in at least one geometric arrangement of P_{rh}. To reduce computation in the optimization phase, we neglect all q_c which have no contribution to the cost function (1).

In the optimization step *(see algorithm 2)* we define S as the set of all possible camera state vectors s for a given set, P_{valid}, and a number of cameras n_c.

$$S = \{s \in {P_{valid}}^{n_c} : \textit{no repetition of } s\} \qquad (21)$$

In this work we used a single camera type and the order of q_c in vector s does not matter because each sensor has equal visibility for the same q_c. Consequently, the cardinality of S can been described by the binomial coefficient:

$$|S| = \binom{|P_{valid}|}{n_c} \qquad (22)$$

During optimization we raise the number of cameras until complete human visibility is reached. We are looking for the minimum cost in which the human is completely visible which is expressed as $|HS_s| = 0$ for all positions in P_{rh}.

The optimisation step implies that the human will be visible for a finite number of cameras, however if the human is covered behind a static obstacle (e.g. a pillar) that is not the case. In the future we will introduce a second criterion to check human visibility compared to the least number of sensors. The optimisation will be stopped if an additional sensor does not improve human visibility and the solution with the lowest cost from the previous number of sensors will be regarded as optimum.

Data: P_c, P_r, P_h
Result: P_{valid}

$P_{valid} = \emptyset$;
foreach $q_c \in P_c$ **do**
  $qcvalid$ = false;
  foreach $q_{r,t} \in P_r$ **do**
    build ${}^0T_{mr}$ from $q_{r,t}$;
    adjust E according to ${}^0T_{mr}{}^{-1}$;
    adjust RM_t according to $q_{r,t}$;
    assign RM_t to R;
    find ${}^{mr}T_c$ from RM_t and $q_{r,t}$;
    ray trace EDM for current ${}^{mr}T_c$;
    foreach $q_{h,t} \in P_h$ **do**
      find ${}^{mr}T_h$;
      adjust H according to ${}^{mr}T_h$;
      ray trace HDM for current ${}^{mr}T_c$;
      determine *HS* and *FA*;
      add *HS* and *FA* to *HStemp* and *FAtemp*;
      if $H \neq HS$ **then**
        $qcvalid$ = true;
      end
    end
  end
  if $qcvalid$ **then**
    add q_c to P_{valid};
    store *HStemp* and *FAtemp*;
  end
end

Algorithm 1: Initialization step.

6 EXAMPLES OF OPTIMAL PLACEMENTS

We have performed two experiments with increasing complexity. Both experiments were performed with a depth camera model of 64×50 pixel resolution and $30^\circ \times 40^\circ$ FOV. The robot centred workspace had dimensions of 6 m in width and length and 3 m in height. It was decomposed into 131.072 cells having an equal edge length of 9.375cm. We used a fast voxel traversal algorithm for ray tracing (Amanatides and Woo, 1987) because our environment model consists of an occupancy grid and we are not interested in very accurate depth images, instead we focus on small computational cost. All experiments were performed on a workstation equipped with an Intel Xeon W3680 and 18GB of ram. The ray tracing and cost function evaluation was performed using CUDA on a NVidia GTX670 device with 1536 Cuda cores.

In the first experiment we evaluated all possible

Data: P_{valid}
Result: s_{opti}
$n_c = 1$;
$mincosts = MAX$;
while *true* **do**
  $S = (P_{valid})^{n_c}$;
  abort = false;
  foreach *element s of S* **do**
    $costs = J(s)$;
    if $|HS_s| = 0$ **then**
      abort = true;
    end
    if $costs < mincosts$ **then**
      $mincosts = costs$;
      $s_{opti} = s$;
    end
  end
  if *abort* **then**
    break;
  end
  n_c++;
end

Algorithm 2: Optimization step.

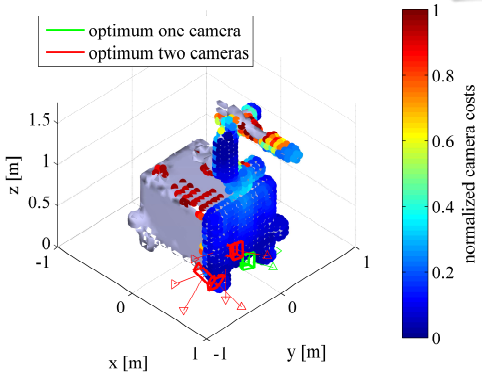

Figure 3: Results for multiple sensors, one human robot position.

camera positions and orientation for a single robot human position pair. The second experiment evaluates a path scenario displayed in figure 4 in which the human crosses the robot path. In the second scenario we were just able to compute the cost function for a single camera position because of limited computational resources.

6.1 Multiple Sensors, one Human Robot Position

The geometric arrangement of the first experiment is illustrated in figure 1. The human is standing at 2.5

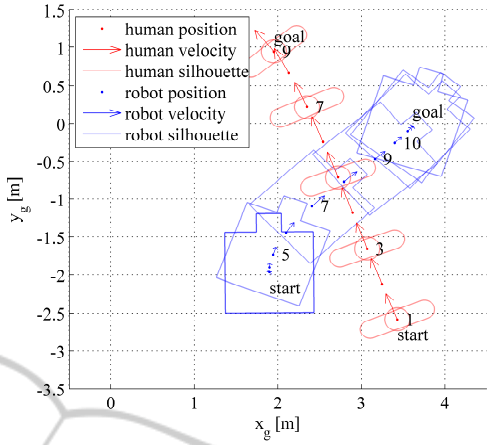

Figure 4: Human and robot path.

m in the positive x direction in front of the robot. We have evaluated 2,590 possible camera positions on the robot and sampled each angular degree of freedom 10 times, thus the resulting number of samples is 2,590,000. The minimum cost value for all angles of the same Cartesian position of P_{valid} is illustrated by the colour value of the Cartesian point in figure 3. $|P_{valid}|$ was 241,660, thus the amount of sample points for the multi camera optimisation was reduced to 10.71 % of the original amount of position and orientations. This reduction is important as the number of samples to be evaluated for each camera is increased according to the binomial coefficient of equation 22.

The complete human visibility was found with two cameras after 176 sec, as displayed in figure 3. The optimal solution for one camera is close to the ground as the distance to the human body increases, which results in a large volume covered. The second camera for the optimal solution for two cameras is almost overlapping with the optimal solution for a single camera because the second camera maximizes the secondary criterion, the visibility of free area.

6.2 One Sensor, Multiple Human Robot Positions

In the second experiment we evaluated the camera costs for a single camera in a dynamic scenario with human and robot trajectories displayed in figure 4. The robot trajectory was generated using the original path planner from the robot manufacturer[1] giving equivalent paths in real world. The human path was generated using the motion planner for the holonomic

[1]http://www.neobotix-robots.com/industrial-robot-mm-800.html

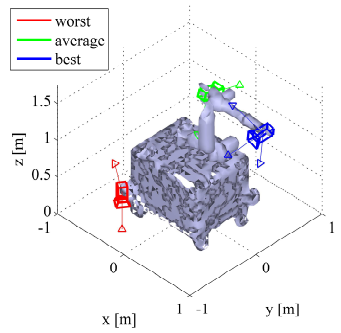

Figure 5: Best, medium and worst camera position on the robot.

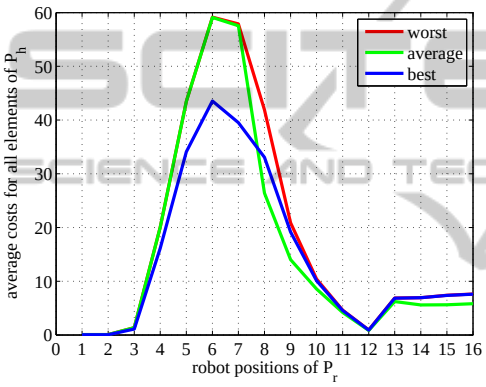

Figure 6: Best, medium and worst camera position on the robot.

mobile robots of Siegwart (Siegwart et al., 2011) taking the human motion constraints of Brogan (Brogan and Johnson, 2003) into account. The paths were sampled at a time step of 1 sec. The cardinality of P_{rh} was 156 consisting of 9 human and 16 robot positions from which 143 configurations were without collisions and considered in the cost function. All 2,590,000 camera positions were evaluated in 366 sec utilizing the cost function from equation 20 for multiple human robot positions.

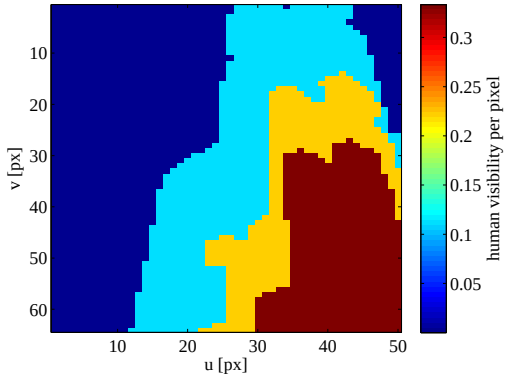

Figure 7: Human visibility per pixel for the best camera position.

The worst, average and best camera positions for the second scenario are illustrated in figure 5. The average costs over all human positions for a single robot position are compared in figure 6. All three camera positions have their minimum at $p_{r,1}$ and $p_{r,12}$ respectively and their maximum at $p_{r,6}$. The robot does not move instantly, thus the robot translational speed is zero followed by a clockwise turn. Consequently the KSDF is almost zero at $p_{r,1}$ which causes very low costs for all sensor positions. The maximum cost value is reached at $p_{r,6}$ because the robot is almost intersecting the human path, therefore the human robot distance is very low which causes a very high KSDF value for human shadowed cells. The second minimum at $p_{r,12}$ is caused by a second stop before turning counter clockwise which results in better performance for the average position as it is now turned towards the human path. All three camera positions perform equally if the human is not seen by any of them because the free area behaves as a secondary criterion as explained in section 5.

The human visibility at $p_{r,6}$ for all human positions of the best camera position can be seen in figure 7. The best camera position covers the first half of the human path, nevertheless the human is just visible in one third of the human positions. Multiple sensors are required to increase human visibility and improve collision avoidance.

7 CONCLUSION

A novel approach to the placement of multiple depth sensors on a mobile robot for human robot cooperation is presented in this work. We have quantified the danger of collision between a human and a robot in pre-determined task positions and combined this with maximum human visibility to obtain a numerically solvable cost function. Experiments have shown promising results for optimal camera placement using the cost function, but in future work these need to be further evaluated by running more complex scenarios in simulation before moving to real world scenarios. Even simple scenarios with small robot and human trajectories require 4.5 days for single camera optimisation with poor human visibility. However, real world cooperative human robot scenarios consist of numerous tasks, task execution orders and varying robot and human trajectories. Even

though computation is offline, exhaustive enumeration of a large number of potential robot, sensor and human positions combined with finite computational resources demands further research into both the candidate function for single and multi-camera placement, and an optimisation strategy that does not require such exhaustive enumeration. Promising approaches are genetic algorithms (Dunn et al., 2006) or simulated annealing (Mittal and Davis, 2008) for highly non-linear optimisation to reduce the number of camera state vectors s to be evaluated. Fast optimisation strategies enable online robot path planning for full human visibility and best collision avoidance performance.

ACKNOWLEDGEMENTS

The authors would like to acknowledge the support of the University of Applied Sciences, Ingolstadt and the Engineering and Physical Research Council (EP/J015180/1 Sensor Signal Processing).

REFERENCES

Amanatides, J. and Woo, A. (1987). A fast voxel traversal algorithm for ray tracing. In *Proceedings of EUROGRAPHICS*.

Angerer, S., Strassmair, C., Staehr, M., Roettenbacher, M., and Robertson, N. (2012). Give me a hand. In *Technologies for Practical Robot Applications (TePRA), 2012 IEEE International Conference on*.

Bodor, R., Drenner, A., Schrater, P., and Papanikolopoulos, N. (2007). Optimal camera placement for automated surveillance tasks. *Journal of Intelligent & Robotic Systems*.

Brogan, D. C. and Johnson, N. L. (2003). Realistic human walking paths. In *Computer Animation and Social Agents, 2003. 16th International Conference on*.

Chen, X. and Davis, J. (2008). An occlusion metric for selecting robust camera configurations. *Machine Vision and Applications*.

De Luca, A., Albu-Schaffer, A., Haddadin, S., and Hirzinger, G. (2006). Collision detection and safe reaction with the dlr-iii lightweight manipulator arm. In *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*. IEEE.

Dhillon, S. S., Chakrabarty, K., and Iyengar, S. S. (2002). Sensor placement for grid coverage under imprecise detections. *Information Fusion, 2002. Proceedings of the Fifth International Conference on*.

Dunn, E., Olague, G., and Lutton, E. (2006). Parisian camera placement for vision metrology: Evolutionary Computer Vision and Image Understanding. *Pattern Recognition Letters*.

Ebert, D. M. and Henrich, D. D. (2002). Safe human-robot-cooperation: Image-based collision detection for industrial robots. In *Intelligent Robots and Systems, 2002. IEEE/RSJ International Conference on*. IEEE.

Fischer, M. and Henrich, D. (2009). 3D collision detection for industrial robots and unknown obstacles using multiple depth images. *Advances in Robotics Research*.

Flacco, F. and De Luca, A. (2010). Multiple depth/presence sensors: Integration and optimal placement for human/robot coexistence. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*. IEEE.

Graf, J., Puls, S., and Wörn, H. (2010). Recognition and understanding situations and activities with description logics for safe human-robot cooperation. In *COGNITIVE 2010, The Second International Conference on Advanced Cognitive Technologies and Applications*.

Haddadin, S. (2013). *Towards Safe Robots: Approaching Asimov's 1st Law*. Springer Publishing Company, Incorporated.

Henrich, D. and Gecks, T. (2008). Multi-camera collision detection between known and unknown objects. In *Distributed Smart Cameras, 2008. ICDSC 2008. Second ACM/IEEE International Conference on*. IEEE.

Kulic, D. and Croft, E. (2007). Affective state estimation for human–robot interaction. *Robotics, IEEE Transactions on*.

Lacevic, B. and Rocco, P. (2010). Kinetostatic danger field - a novel safety assessment for human-robot interaction. In *Intelligent Robots and Systems (IROS), 2010*.

Lenz, C. (2012). Fusing multiple kinects to survey shared human-robot-workspaces. Technical report, Technical Report Technical Report TUM-I1214, Technische Universität München, Munich, Germany.

Mittal, A. and Davis, L. (2008). A General Method for Sensor Planning in Multi-Sensor Systems: Extension to Random Occlusion. *International journal of computer vision*, 76(1):31–52.

Nikolaidis, S., Ueda, R., Hayashi, A., and Arai, T. (2009). Optimal camera placement considering mobile robot trajectory. In *Robotics and Biomimetics, 2008. IEEE International Conference on*. IEEE.

O'rourke, J. (1987). *Art gallery theorems and algorithms*, volume 1092. Oxford University Press Oxford.

Siegwart, R., Nourbakhsh, I. R., and Scaramuzza, D. (2011). *Introduction to autonomous mobile robots*. MIT Press.

Sisbot, E., Marin-Urias, L., Broquere, X., Sidobre, D., and Alami, R. (2010). Synthesizing robot motions adapted to human presence. *International Journal of Social Robotics*.

Yao, Y., Chen, C.-H., Abidi, B., Page, D., Koschan, A., and Abidi, M. (2008). Sensor planning for automated and persistent object tracking with multiple cameras. In *Computer Vision and Pattern Recognition, 2008. IEEE Conference on*. IEEE.