3rd IFToMM International Symposium on Robotics and Mechatronics
2 – 4 October 2013, Singapore

ACTIVE GAZE STRATEGY FOR REDUCING MAP UNCERTAINTY ALONG A PATH

M. BRANDÃO*
Grad. School of Advanced Science and Engineering, Waseda University, Japan,
mbrandao at fuji.waseda.jp

R. FERREIRA
Institute for Systems and Robotics, Instituto Superior Técnico, Portugal,
ricardo at isr.ist.utl.pt

K. HASHIMOTO
Faculty of Science and Engineering, Waseda University, Japan,
k-hashimoto at takanishi.mech.waseda.ac.jp

J. SANTOS-VICTOR
Institute for Systems and Robotics, Instituto Superior Técnico, Portugal,
jasv at isr.ist.utl.pt

A. TAKANISHI
Faculty of Science and Engineering, Waseda University, Japan,
contact at takanishi.mech.waseda.ac.jp

Robots depend on a world map representation in order to navigate on it. Only a part of the space around the agent can be sensed at each time and so measures must be taken in order to reduce the uncertainty of this map and likelihood of collision. In this work we propose the use of a probabilistic occupancy grid to guide active gaze of the robot on the "walk to target" task. A map uncertainty measure is proposed, as is a method for choosing gaze points along the robot's computed trajectory to anticipate the need for trajectory changes.
Gaze points are chosen from the whole space volume the robot will traverse. Then, robot trajectories are computed directly on the probabilistic map in order to drive the robot towards free-space areas of high confidence. A preliminary evaluation of the approach is done on a real scenario using the humanoid robot KOBIAN for the preparatory gaze exploration task necessary for safe trajectory planning to a target.

INTRODUCTION

World representation is an important part of perception for the purpose of interacting with the agent's environment. One example of such interactions is navigation, which is a problem tied with that of modeling, or mapping, the world around the agent.
On a typical mapping task the agent navigates through the map, updating the map or its own position in the map according to odometry measurements which naturally deteriorate with time. Solving this localization problem at the same time as mapping the world is known as the problem of Simultaneous Localization and Mapping (SLAM) [1]. Sparse points in the world are defined to

the agent itself and self-localization is made with respect to these points. A sparse world representation is not, though, suitable for motion planning.
Occupancy grids [2], on the other hand, are metric world representations where the world is divided into a grid in which cells have certain physical dimensions and are marked as occupied or free (or with a probability of occupation). This kind of representation is adequate for many motion planning purposes. Inherent to any of these methods is the assumption that the world is static and does not change in time. If one wants to deal with moving obstacles further processing is needed, such as segmentation of these objects in the sensor and estimation of their motion model.

Humanoid robots that have been developed usually mimic humans both in terms of body kinematics and sensory systems. As a consequence, stereo vision is usually used as a sensor for grid mapping. That brings additional challenges to the navigation task: cameras may have a very limited field of view (such as 60 degrees) compared to that of a human (almost 180 degrees). For this reason robots should, more than humans, actively adopt gazing strategies in order to still guarantee a high coverage of space and identify obstacles which sometimes fall out of the robot's field of view.
However, most implementations of humanoid vision-based navigation forget to actively deal with uncertainty of the world representation along the space that will be traversed. Instead, the focus of active vision goes mainly to the tasks of localization and exploration. In [3] [4] active gaze was used in order to improve localization performance by choosing visual landmarks to fixate and track according to the landmarks' Kalman filter innovation covariance. SLAM's Kalman filter covariances were also the object of minimization in [5] and global trajectories were planned which maximize the overall map's quality. Many works focus also on the exploration problem in order to map an entire room or region in the most efficient way [6] [7] [8]. In [9] the localization and obstacle avoidance task compete as gaze targets with a decision based on Utility Theory. Obstacle positions are known, though, and the competing gaze targets are either minima between trajectory and obstacles or landmarks. In [10] a humanoid robot learns in simulation the optimal camera movement strategy for the obstacle avoidance and reaching a target task. A neural network that takes into consideration only gray-levels of the image to decide on camera motion and walking speed is used.

Contribution
In this work we look into two problems: (1) a representation of the world around a robotic agent and (2) actively minimizing the uncertainty of this representation along the robot's trajectory through gaze. This is accomplished through a probabilistic occupancy grid to which a measurement of "space occupation" uncertainty is inherently tied. With the proposed systematic approach, the robot can effectively plan its motion in an unknown environment, gazing at where it is crucial to have a good map: where it plans to walk.
The contributions of the paper are:

- A strategy based on active gaze for reducing the map uncertainty of the space volume the robot will traverse. This guarantees safety of the executed path.
- We consider the mapping problem from an egocentric perspective, concentrating on a limited region surrounding the agent. This region can be described in high resolution which is crucial for obstacle avoidance and deciding gazing actions.
- Path planning is computed directly on the probabilistic map in order to drive the robot towards free-space areas of high confidence.
- The proposed framework was implemented on the KOBIAN humanoid robot and tested on a real scenario.

PROPOSAL

Map Generation

We define an egocentric approach to an Occupancy Grid. Our egocentric map of the world is defined in a region around the robot, centered and aligned with the current sensor orientation. The sensor used for mapping is based on stereo vision using two cameras mounted on the robot. For path planning purposes the world representation is made parallel to the walking surface. The map frame of reference is then defined as: origin obtained by projecting the head reference frame in our walking surface, parallel to this surface, and with the same Yaw orientation as the head. Yaw is here defined as the rotation around the vector normal to the walking surface. We refer to this frame as the grid reference frame, defined with respect to the robot's sensor ${}^{G}T_{S}$ or base ${}^{G}T_{B}$.
An Occupancy Grid algorithm assumes that the exact pose of the robot is known. In this work, we do not focus on the localization problem and assume that this pose is given by some sort of sensor fusion or SLAM running in parallel with our method.

The occupancy grid problem is defined as computing the posterior probability $p(m \mid z_{1:t}, x_{1:t})$. Here p represents the probability of occupation, m is the map, $z_{1:t}$ the set of measurements until time t, and x the state of the robot (defined to a global reference frame attached to the world). Cells are usually considered independent. If so the problem can be split into a binary estimation with static state for each cell individually, making it possible the use of a Bayes filter. Probabilities are here represented as a log-odds ratio L which is an elegant way of posing the binary inference problem:

$$L(x) = \log \frac{p(x)}{1 - p(x)} . \qquad (1)$$

A Bayes filter at a certain cell i can be defined in the log-odds form [11] according to

$$L_t^{(i)} = L_{t-1}^{(i)} + \log \frac{p(O^{(i)} \mid z_t)}{1 - p(O^{(i)} \mid z_t)} - \log \frac{p(O^{(i)})}{1 - p(O^{(i)})}, \qquad (2)$$

where $O^{(i)}$ represents the state "occupied" of cell i. In our implementation we assume a prior $p(O^{(i)})=0.5$ (occupied and free cells are equally probable), eliminating the last log of Eq. (2). In the log-odds representation probabilities are defined in the $\mathbb{R}$ space and hence we opt to limit them to the interval [-10,10] which rounds to probability [0.0005,0.9995]. This is in order to avoid L going to infinity and taking too long to adapt to environment changes which may occur. A 3D implementation of the occupancy grid is used since our objective is to gaze at points in the 3D space. This space is divided into N_X, N_Y and N_Z number of cells in each axis.

The implemented algorithm follows 5 steps:

1. Project all 3D points P_S from sensor to grid coordinate frame $P_G = {}^{G}T_S P_S$.
2. Compute $p(O^{(r)} \mid P_G)$ of all cells r on the ray from P_G to sensor position.
3. For each cell i compute $p(O^{(i)} \mid z_t) = \max p(O^{(i)} \mid P_G)$ of all P_G that projected on that cell
4. Compute $L_{t-1}^{(i)}$ from rotation/translation according to the sensor motion with respect to the world ${}^{S(t)}T_{S(t-1)}$, where $S(t)$ is the Sensor's reference frame at time t.
5. Update occupancy probabilities through Eq. (2)

The term $p(O^{(r)} \mid P_G)$ should be set according to the inverse sensor model. Since this is not the focus of the present paper, we follow [8]'s strategy using "occupied" and "free" confidence priors representing how much the measurements are trusted for each of the states. As in [8], we set $p(O^{(r)} \mid P_G)$ of the cell where P_G projects to $P_{occupied}=0.7$ and cells until the sensor to $P_{free}=0.4$.

Trajectory planning

The generated grid can be projected into a 2D top-view map where each cell's value corresponds to the maximum probability of occupation along the vertical axis. A trajectory to the target can then be computed such as to minimize the occupancy probability along this path.

In this work we opted for an A* approach [12] to the search problem. We chose this method for its simplicity, although other more efficient approaches could be used. We use a set of predefined robot motions to build the search graph, adjusted to the motion capabilities and limitations of our robot such as maximum turning angle. The path was planned in (angle, length) space due to details in the interaction with the robotic platform. The graph is searched such that only nodes with occupancy probability lower than $P_{walkable}$ are explored, and the cost associated to a certain motion is set as to grow exponentially with occupancy probability of the cells it traverses. The Euler distance was used as a heuristic for the cost to the goal. The result is exemplified in Fig. 2.

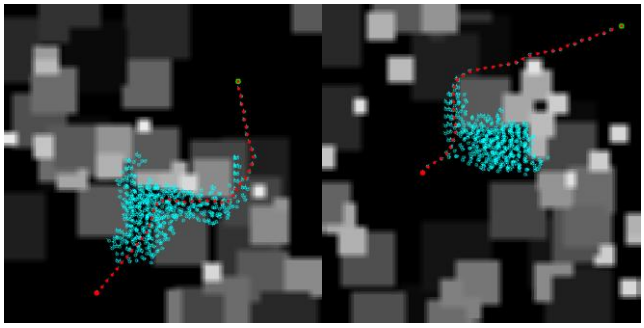

Figure 2. Two examples of simulated scenarios with regions of different occupancy probability. The brighter the pixel the higher the probability. Trajectory nodes explored (closed list of the A* algorithm) are in cyan and final solution in red. With this approach we look for minimum cost trajectories preferring regions with low occupation probability.

For planning purposes it is usually useful to consider the robot as a point in the grid and obstacles are dilated according to the robot's dimensions. Here we keep the grid probabilistic, without classifying cells into occupied or free. Therefore the grid can be dilated by taking for each grid cell the maximum of occupancy probability in the robot's area around that cell.

Map uncertainty along trajectory

Cells in our map are represented as binary random variables and uncertainty of these variables can be measured using its Shannon Entropy value for a binary variable:

$$h(X,Y,Z) = -P(X,Y,Z)\log P(X,Y,Z) - Q(X,Y,Z)\log Q(X,Y,Z), \qquad (4)$$

where $P(X,Y,Z)$ is the probability of occupation of that cell and $Q(X,Y,Z)=1-P(X,Y,Z)$. For each (X,Y) on the grid we compute the total entropy

$$H(X,Y) = \sum_{Z} h(X,Y,Z) \,. \tag{5}$$

As the map is being built, cells can be queried for uncertainty, and high uncertainty points can be chosen as gaze targets in order to reduce map uncertainty and thus collision chances. An example of the map is shown in Figure 3. We assume general 2D trajectories with some velocity profile and look for uncertainty maxima along time windows of the trajectory (Fig. 4). By looking for these maxima in time instead of space, we take into account the robot's velocity. Thus, we solve

$$t^*(w) = \underset{t\in[d^*w,d^*(w+1)]}{\arg\max} \; H(c(t)) \,, \tag{6}$$

with H(c(t)) being the total entropy at point c(t) in the trajectory parameterized in time. c(t*(w)) are the chosen gaze points defined on the 2D grid map along the w^{th} window of trajectory, and d is the window duration.

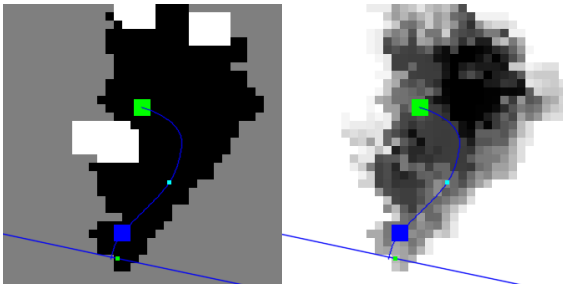

Figure 3. Left: 2D top-view of the Occupation Grid. Black is occupancy P=0 and white P=1; trajectory chosen offline and represented in blue; big green square is the target; small is the sensor; and squares along the trajectory represent uncertainty maxima. Right: 2D representation of map uncertainty H(X,Y), white is max. uncertainty, black is minimum.

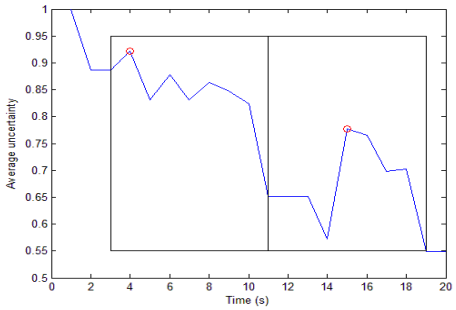

Figure 4. Average uncertainty at each point along the trajectory with respect to time. Boxes represent windows for search of maxima and red dots represent those maxima. Trajectory windows start at 3s to avoid regions too close to the robot.

Results in Fig. 4 show that function H(c(t)) is predictably higher as points get close to the sensor's cone origin, since points on the vertical direction are less and less sensed.

Where to gaze

To decide which of the generated points to gaze at, we select the first one with uncertainty higher than a threshold $TH_{uncertain}$, thus giving priority to closer points as argued previously. The least priority point (i.e. the one gazed at after the robot is certain about its trajectory) is the target.

The chosen gaze point in the 2D uncertainty map is then mapped back to the 3D world. The choice of the gaze point's height is made by computing a center of maximum uncertainty. One option would be to take the maximum uncertainty $argmax(h_{X,Y}(Z))$. However, for $h_{X,Y}(Z)=a$ (constant), this would be equal to one of the limits of the function. Choosing the "center of mass" of $h_{X,Y}(Z)$ could lead to gaze points staying forever in the middle of two maxima on the function limits. We suggest a combined approach by segmenting $h_{X,Y}(Z)$ into connected regions (Fig. 5) and gazing at the center of mass of the region with highest total uncertainty.

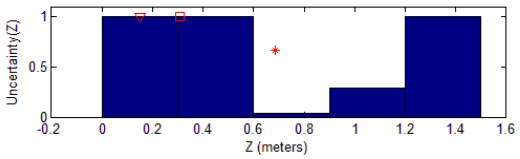

Figure 5. Uncertainty along the vertical axis during one experiment with the robot. Maximum uncertainty points are marked with a triangle (gaze to borders); center of mass of the uncertainty with a star (gaze to a low uncertainty region); and our choice with a square.

Tracking of the gaze target is done using only neck joints. The same method can equally be used with eye saccades although one must have in mind, of course, the effects on image rectification for stereo vision in that case, which will render a narrower field of view to the system.

RESULTS

Robotic platform and experimental setup

This research was implemented on the biped humanoid robot platform KOBIAN [13]. KOBIAN was built with human robot interaction in mind, it is 1.4m tall, weighs 62kg and has a total 48 DoF. The vision system uses two CMOS cameras working at a 30Hz acquisition rate.

Camera images were used at a 320x240 pixel resolution and a standard Block-Matching algorithm implementation present in OpenCV [14] was used to generate disparity images.

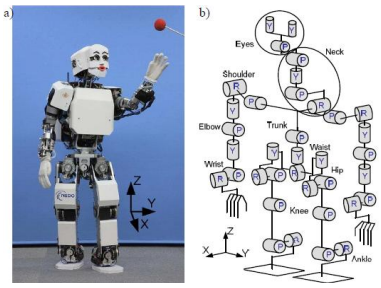

Figure 6. The humanoid robot KOBIAN used for this work (a), its kinematic structure (b).

Since the Grid Occupancy algorithm is of complexity $O(N_X*N_Y*N_Z)$, the choice of these values can change performance drastically. In these experiments they were set according to the physical dimensions of the robot. Having in mind the average step size of the robot, 0.20 meters, we decided on the dimensions [0.15x0.15x0.30] meters for the cell size and $(N_X,N_Y,N_Z)=(61,61,5)$. We assume 2D trajectory planning of the robot and hence, for performance reasons, decided to lower the resolution along Z comparing to the other axes.

The tested scenario is as follows: the robot stands in a room looking forward, having a target where it has to walk to, fixed in the world (3m ahead, 2m to the left). Between the robot and the target, some common obstacles such as chairs were placed. For simplified collision detection, occupied cells were dilated taking into account the robot's dimensions (approximately a 60x60cm square on the 2D plane) – so that the robot can be represented as a single cell in the map. The generated trajectories have a constant velocity of 1cell/sec (roughly equivalent to 1 step per second on the KOBIAN robot). Gaze targets will be searched along this path.

Experiment results

The proposed algorithm is started once the robot is on the floor, successfully generating gaze targets along the trajectory (Fig. 7). Chosen gaze targets were at first along uncertain regions close to the robot (uncertain due to the cone shape of the stereo sensor, as discussed in the previous section) and then towards the walking goal. All was generated online and automatically without human intervention. The duration of the experiment was approximately 1 minute.

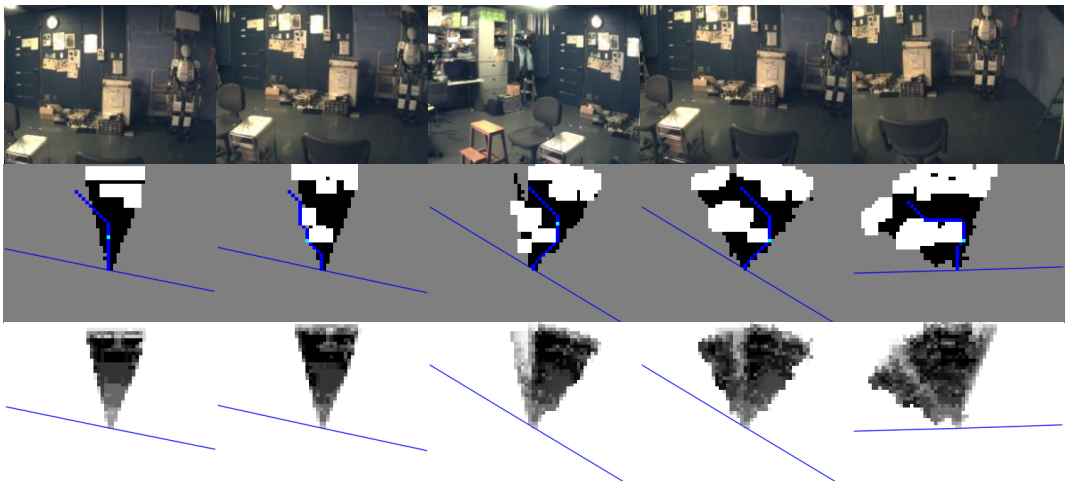

Figure 7. Results of the proposed gazing strategy. From top to bottom: 1) Right camera image; 2) Occupancy map (projected to 2D and dilated to robot dimensions). Egocentric representation: vertical direction in the image corresponds to current sensor direction. The thin blue line indicates the orientation of the robot base with respect to the sensor (imagine a line piercing the robot's waist from the left to the right). Black is probability of occupation P=0 and white P=1, generated trajectory blue, and gaze target cyan point; 3) Resulting uncertainty of the map. White means max. uncertainty, black min. uncertainty. From left to right: frames 112, 208, 310, 417, 523.

At first, an obstacle is detected that leads to a trajectory around its left. After gazing at this trajectory's high uncertainty regions, a new obstacle is found and the trajectory updated. Due to the narrow field of view of the stereo sensor, regions close to the floor are not sensed in the beginning (the robot looks straight ahead). The generated gaze targets are as such lower than the starting one.

We then compare the result, in the exact same scenario, of adopting a "gaze at target" approach to mapping and planning. We conducted an experiment where the robot starts in the same condition as in the previous one but then looks straight to the target without any previous gazing points. The result (Fig. 8) is that some obstacles that were sensed with active gazing are not sensed anymore because: they are either 1) too low (in height) with respect to the walk target. This is a typical situation for human-sized humanoids. Or 2) they extend to an area not only restricted to the straight line robot-target. In this case the trajectory would actually lead to a collision.

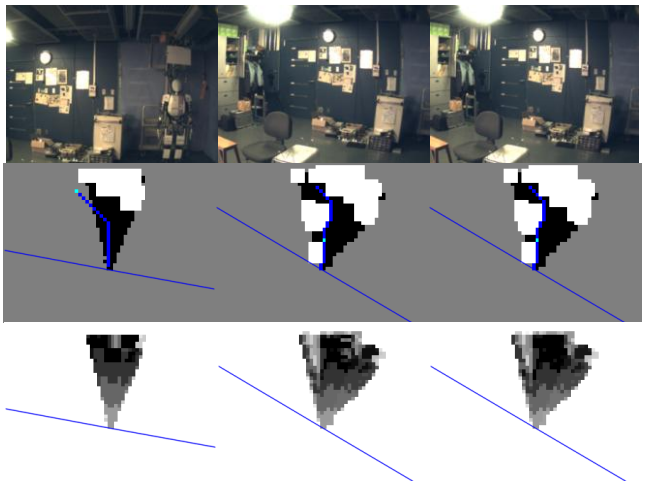

Figure 8. Results of "gaze at target" strategy. From top to bottom: 1) Right camera image; 2) Map occupancy 3) Map uncertainty. From left to right: frames 101, 207, 316 – after this point the map does not change anymore.

On a final experiment we quantitatively compare the results of our active gaze strategy with a "random-gaze" strategy. Gaze orientations were defined in pan/tilt space taking into consideration robot joint limits and sampled randomly in that space at the same frequency as our active strategy. To show the advantages of the proposed method, we took the best-case of the random approach where it "accidentally" looks to free space where a trajectory can be executed. The scenario is similar to the one presented on the previous experiments and results are shown in Fig. 9.

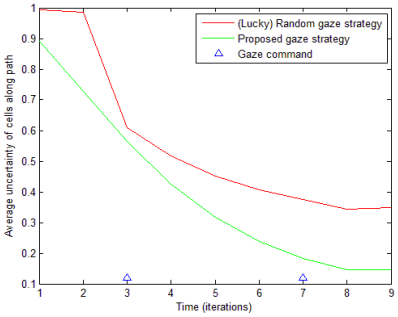

Figure 9. Average entropy of cells along planned path as a measure of path "safety". Two strategies are compared: random gaze and proposed active gaze system. Time t in algorithm iterations. Gazing executed every 4 iterations. First gaze at t=3.

In both experiments the robot starts in the same configuration and so the same path is planned and same average entropy achieved until the first gaze command is executed (t=3). At this point, as we can see from Fig. 9, even on a lucky gaze target for the random gaze strategy, uncertainty is lowered considerably slower when compared to the proposed approach. At t=7 our approach has already achieved uncertainty 0.15 while random is still 0.35.

CONCLUSIONS

We proposed and tested on a humanoid robot, KOBIAN, an active gazing strategy based on uncertainty maps. Although active gaze has been proposed before to maximize localization information, this goal could conflict with that of navigation in an unexpected obstacle-enhanced environment. In the system we propose, gaze target is updated every few iterations to a new one along the planned walking trajectory – which lowers uncertainty along the trajectory and thus collision probability. The proposed system was tested on a real scenario, recognizing obstacles after gazing actions were taken, and updating the map along the walking trajectory. We believe the proposed uncertainty representation is extremely useful for navigation purposes and practical due to its simplicity and lack of processing requirements in case a grid-like representation is chosen as the world mapping method.

ACKNOWLEDGEMENTS

This study was conducted as part of the Research Institute for Science and Engineering, Waseda University, and as part of the humanoid project at the Humanoid Robotics Institute, Waseda University. It was also supported in part by RoboSoM project from the European FP7 program (Grant agreement No. 248366), MEXT/JSPS KAKENHI (Grant Number: 24360099), Global COE Program "Global Robot Academia", MEXT, Japan, SolidWorks Japan K.K., and DYDEN Corporation whom we thank for their financial and technical support.

REFERENCES

[1] Smith, R. C. and Cheeseman, P., "On the representation and estimation of spatial uncertainty", The international journal of Robotics Research, Sage Publications, 5, 56-68, 1986.

[2] A. Elfes, “Using occupancy grids for mobile robot perception and navigation,” Computer, vol. 22, no. 6, pp. 46–57, 1989.
[3] D W Murray and A J Davison, “Mobile Robot Localization using Active Vision”, Proc 5th European Conference on Computer Vision, II, pp 809-825, 1998.
[4] Lidoris, G. et al, "Information-based gaze direction planning algorithm for SLAM", 6th IEEE-RAS International Conference on Humanoid Robots, 302-307, 2006.
[5] Sim, R., Roy, N., "Global A-Optimal Robot Exploration in SLAM," Proceedings of the IEEE International Conference on Robotics and Automation, 2005, pp. 661-666, April 2005.
[6] Strand, M. and Dillmann, R., "Using an attributed 2D-grid for next-best-view planning on 3D environment data for an autonomous robot", International Conference on Information and Automation 2008, 314-319, 2008.
[7] Yamauchi, B., "A frontier-based approach for autonomous exploration", 1997 IEEE International Symposium on Computational Intelligence in Robotics and Automation, 146-151, 1997.
[8] R. Shade and P. Newman, “Choosing where to go: Complete 3d exploration with stereo,” in Robotics and Automation (ICRA), 2011 IEEE International Conference on. IEEE, 2011, pp. 2806–2811.
[9] Seara, J. F. and Schmidt, G. (2004). Intelligent gaze control for vision-guided humanoid walking: methodological aspects. Robotics and Autonomous Systems, 48(4), 231-248
[10] Suzuki, M., Gritti, T. and Floreano, D., "Active vision for goal-oriented humanoid robot walking", Creating Brain-Like Intelligence, Springer, 303-313, 2009.
[11] S. Thrun, W. Burgard, and D. Fox., “Probabilistic Robotics”, Intelligent Robotics and Autonomous Agents series, Intelligent robotics and autonomous agents, The MIT Press, August 2005.
[12] Hart, P. E., Nilsson, N. J., Raphael, B. , "A Formal Basis for the Heuristic Determination of Minimum Cost Paths", IEEE Transactions on Systems Science and Cybernetics , vol.4, no.2, pp.100-107, July 1968.
[13] N. Endo, S. Momoki, M. Zecca, M. Saito, Y. Mizoguchi, K. Itoh, and A. Takanishi, “Development of whole-body emotion expression humanoid robot,” in Proc. IEEE Int. Conf. Robotics and Automation ICRA, 2008, pp. 2140–2145.
[14] Bradski, G. (2000). The OpenCV Library. Dr Dobbs Journal of Software Tools, 25(11), 120-126.