

Learning the reachable space of a humanoid robot: a bio-inspired approach*

Lorenzo Jamone¹, Lorenzo Natale², Kenji Hashimoto¹, Giulio Sandini² and Atsuo Takanishi^{1,3}

Abstract—In this paper we describe how a humanoid robot can learn a representation of its own reachable space from motor experience: a Reachable Space Map. The map provides information about the reachability of a visually detected object (i.e. a 3D point in space). We propose a bio-inspired solution in which the map is built in a gaze-centered reference frame: the position of a point in space is encoded with the motor configuration of the robot head and eyes which allows the fixation of that point. We provide experimental results in which a simulated humanoid robot learns this map autonomously and we discuss how the map can be used for planning whole-body and bimanual reaching.

I. INTRODUCTION

Our society looks at robots as future helpers for humans: they could assist elderly people, cooperate with us in the execution of tedious and hard works, and even replace us for the accomplishment of the most dangerous tasks. This long term goal is pushing robotic researchers to build more and more complex robots, especially humanoids, which exploit state-of-the-art technologies for their structure and actuation, showing an increasing number of degrees of freedom and sensors [1], [2]; these robots should be able to cope with the unstructured environment in which humans daily live and act. In particular, they should be able to reach for objects using the arms or even the whole body, eventually ending up grasping and using such objects. An important issue associated with these behaviors is the definition of the space that the robot can reach, i.e. the reachable space or workspace. In general, if an accurate model of the system is available, analytical or geometric methods can be used to analyze and obtain the robot reachable space. However, to build analytical model of current humanoid robots is becoming a more and more difficult task, due to their increasing complexity. Learning techniques offer an interesting solution if insufficient analytical knowledge is available, and seem even mandatory as long as humanoids are supposed to become completely autonomous and gain knowledge through their own experience.

This work explores the possibility of learning a representation of the robot reachable space from motor experience:

*This work was partially supported by RoboSoM project from the European FP7 program (Grant agreement No.: 248366).

¹L. Jamone, K. Hashimoto and A. Takanishi are with the Faculty of Science and Engineering, Waseda University, Tokyo, Japan
lorejam at liralab.it , contact at takanishi.mech.waseda.ac.jp

²L. Natale and G. Sandini are with the Department of Robotics, Brain and Cognitive Sciences, Italian Institute of Technology, Genoa, Italy
lorenzo.natale,giulio.sandini at iit.it

³A. Takanishi is with the Humanoid Robotics Institute, Waseda University, Tokyo, Japan

a Reachable Space Map. We have already introduced this idea in a recent work (not published yet) in which only very preliminary results are provided. The robot acquires autonomously a kinematic model of its body through exploration of the motor space (i.e. motor babbling); the knowledge of this model allows to produce training data to learn the Reachable Space Map. After learning, the robot can use the map to estimate the reachability of a visually detected object before starting the reaching movement. Furthermore, the map can also be exploited to plan body movements that would make the subsequent reaching action successful (e.g. bending or rotating the waist, bending the knees, walking toward the object).

The main aspect which makes our solution innovative with respect to previous works is the use of a gaze-centered reference frame to describe the robot workspace. Indeed, the position of a point in space can be defined in different ways, the most common one being the Cartesian position with respect to fixed reference frame, either placed somewhere in the environment or on the robot body. Conversely, here we propose to encode positions in space with motor coordinates: in particular, the position of a point in space is defined by the motor positions of the head and eyes when the robot is fixating that point, or in other terms by the *gaze configuration*. This choice is inspired by converging evidence both in monkeys and humans, that will be discussed in Section II. Then in Section III we review the studies in which this idea has been applied to robotics and those which address the generation of the reachable space of humanoid robots. The details of our implementation are described in section IV. Finally, simulation results obtained with the dynamic simulator of the humanoid robot iCub are presented in section V, while in Section VI we discuss how the Reachable Space Map can be used for planning whole-body and bimanual reaching. In section VII we report our conclusions.

II. A BIO-INSPIRED APPROACH

Experiments on monkeys reveal that the parietal reach region (PRR) of the posterior parietal cortex (PPC) codes the location of reaching targets [3], [4]. This information is encoded explicitly in a gaze-centered, eye-fixed frame of reference, and is updated during eye movements [4], [5] as opposed to a head or body-centered frame, which would be independent of eye movements [6].

In humans, evidence from fMRI (functional Magnetic Resonance Imaging) recordings suggest that spatially contiguous neuronal populations operate in different frames of reference,

supporting sensorimotor transformations from gaze-centered to body-centered coordinates [7]. In particular, the posterior parietal cortex dynamically updates the spatial goals for action in a gaze-centered frame [8]. Behavioral studies enforce the hypothesis of the presence of a gaze-centered frame of reference for the control of pointing [9]–[12] and reaching [13], even in the case of whole-body reaching [14]. Furthermore, neurophysiological evidence based on TMS (Transcranial Magnetic Stimulation) measurements show that human perception of what is reachable or not relies on motor information [15], [16]. In [15] the task of visually judging the reachability of objects was influenced by the application of TMS on the motor area (facilitation effect), while in [16] recorded motor evoked potentials (MEPs) from hand muscles were found to be higher when visually detected objects were lying within the subjects reachable space, during magnetical stimulation of the primary motor cortex.

III. RELATED WORKS

The idea of encoding positions in space by using motor variables is not new in robotics. Different works show how this solution allows to learn a motor-motor mapping that can be used for reaching control [17]–[19]. This approach has been considered also in recent works [20], [21] in which the authors mention a motor representation of the space surrounding the robot, either referred as “embodied representation of space” or “visuomotor representation of the peripersonal space”. However, none of these works makes this representation explicit, and they only represent the space that the robot can reach (not the space that the robot cannot reach); therefore, it is not clear how this information can be used for planning body movements that facilitate the subsequent reaching action.

Recently numerical methods have been applied in order to build a representation of the reachable space of humanoid robots. In [22] an optimization-based method and the Monte Carlo method are compared: locations in space are associated with a binary information (i.e. reachable / non-reachable) which is stored in a database for later utilization. A richer description of the robot workspace is provided in [23], where reachable points in space are ranked by their *reachability*, a measure indicating the number of possible approaching directions for the arm; on the basis of that measure a *capability map* is built which has a directional structure and can be used to identify good approach directions for grasping objects, as shown in a subsequent work [24]. In [25] the reachability space is represented by a grid of voxels holding information about the success probability of an IK (inverse kinematics) query, and it is used to speed up a randomized IK solver.

In all these works the robot workspace is described with respect to a Cartesian frame of reference (either placed in the world or on the robot). With respect to this solution, representing locations in space using the gaze configuration has several advantages. First, we can easily represent all locations the robot can see, even if not reachable, in a compact map, which is limited by the robot joints limits.

This is not possible using other representations, and in fact other works either encode only reachable locations or create arbitrary limitations around the robot (e.g. bounding box). Second, this representation is directly linked with the visual search: after a visually detected object is fixated the reachability information can be retrieved without the need of any additional transformation. Moreover, the gaze configuration can be used to directly trigger the reaching movement, as described in the literature [17]–[21].

Furthermore, previous works provide a discrete representation of the space (i.e. grid of voxels). Conversely, in our approach the reachable space is approximated using a LWPR (Locally Weighted Projection Regression) neural network [26], which deals with continuous input and output; we will further explain how this is important when using the map for planning body movements. Using LWPR also provides a compact representation that can be queried in real-time to get online evaluations of the reachability of visually detected object; memory occupancy and response time (i.e. the time needed to perform a query) are typically smaller than databases or other neural networks.

IV. PROPOSED SOLUTION

The definition of the reachable space is based on a kinematic model of the robot arm, head and eyes, namely the arm-gaze forward kinematics. Given the arm configuration, the model provides the gaze configuration (head+eyes configuration) which allows fixation of the robot end-effector (i.e. the hand). This model is learned online and autonomously by the robot through motor babbling using LWPR. Then, training data to estimate the reachable space map is obtained by solving a large number of IK queries (i.e. inversion of the learned kinematic model). We propose two different solutions to design the Reachable Space Map: the *Basic Map* and the *Enhanced Map*. While the output of the Basic Map estimates the probability of having a valid IK solution for a given gaze configuration (i.e. it estimates whether the fixated point is reachable or not), the output of the Enhanced Map provides a richer information about the degree of reachability of the fixated point, according to a certain quality measure. The Enhanced Map indicates not only whether the fixated point is reachable or not but also how well the robot can reach for it (in case of reachable locations) or how far from reachable it is (in case of non-reachable locations). Both maps are implemented using a LWPR neural network. Hereinafter we describe the different components of the system in detail.

A. The *iCub Simulator*

The experiments described in the paper were carried on using the *iCub Dynamic Simulator* [27], displayed in Figure 1. We actuate 3 DOFs of the head+eyes system and 4 DOFs of the arm, namely:

$$\begin{aligned} \cdot \mathbf{q}_{gaze} &= [\theta_y \ \theta_p \ \theta_v]^T \in \mathbb{R}^3 \\ \cdot \mathbf{q}_{arm} &= [\theta_{sy} \ \theta_{sp} \ \theta_{sr} \ \theta_e]^T \in \mathbb{R}^4 \end{aligned}$$

where θ_y and θ_p are the neck yaw and pitch angles respectively (pan and tilt of the head), θ_v is the eyes vergence angle,

θ_{sy} , θ_{sp} , θ_{sr} are the shoulder yaw, pitch and roll angles respectively (elevation/depression, adduction/abduction and rotation of the arm) and θ_e is the elbow flexion/extension angle.

The robot joints limits are defined as follows:

$$\begin{aligned} \cdot \mathbf{q}_{gaze}^{min} &= [-55^\circ; -40^\circ; 5^\circ]^T \\ \cdot \mathbf{q}_{gaze}^{max} &= [55^\circ; 40^\circ; 90^\circ]^T \\ \cdot \mathbf{q}_{arm}^{min} &= [-95^\circ; 0^\circ; -20^\circ; 15^\circ]^T \\ \cdot \mathbf{q}_{arm}^{max} &= [10^\circ; 160^\circ; 80^\circ; 105^\circ]^T \end{aligned}$$

Standard image processing techniques are used to segment and to compute the center position of the hand (i.e. the palm, colored in green) and the object (i.e. the red ball).

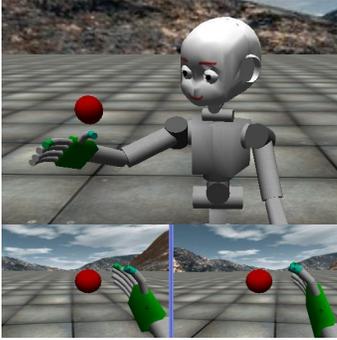


Fig. 1. The dynamic simulator of the humanoid robot iCub. Top: view from outside. Bottom: view from robot cameras (i.e. robot viewpoint).

B. Gaze control

The gaze controller allows the robot to fixate a 3D point in space by moving the head and eyes. We will refer to this point as the “target” (i.e. target of the gazing action) to explain how this controller works. Then, in the rest of the paper, the target can be either the hand or the object, and we use the verbs “to gaze at” or “to fixate” to indicate the activation of this controller.

If the target is visible (i.e. inside the image plane) joints velocities are generated as follows:

$$\dot{\mathbf{q}}_{gaze}(t) = -G\mathbf{x}(t) \quad (IV.1)$$

where $G \in \mathbb{R}^{3 \times 3}$ is a positive definite gain matrix and the position of the target $\mathbf{x} \in \mathbb{R}^3$ is defined as follows:

$$\mathbf{x} = \begin{bmatrix} u_L - u_R \\ (u_L + u_R)/2 \\ (v_L + v_R)/2 \end{bmatrix} = \begin{bmatrix} 1 & -1 & 0 & 0 \\ 1/2 & 1/2 & 0 & 0 \\ 0 & 0 & 1/2 & 1/2 \end{bmatrix} \begin{bmatrix} u_L \\ u_R \\ v_L \\ v_R \end{bmatrix}$$

being u_R and v_R the coordinates of the target on the right image plane and u_L and v_L the coordinates of the target on the left image plane. Indeed, the goal of the controller is to reduce to zero $[u_L \ u_R \ v_L \ v_R]^T$, which entails bringing the target to the center of both cameras (i.e. the fixation point). However, since $v_L = v_R$ (perceived targets have the same vertical position on both images) it is sufficient to reduce to zero $[u_L - u_R \ (u_L + u_R)/2 \ (v_L + v_R)/2]^T$.

If the target is not visible a stereotyped motion strategy (i.e. random left-right and up-down movements of the neck) is

used to detect it; then controller IV.1 is activated.

After fixation is achieved we encode the target position in space using the gaze configuration \mathbf{q}_{gaze} ; since we actuate only 3 DOFs of the head+eyes system the mapping from gaze configuration to target position is unique. If more DOFs are used the redundancy should be solved by the gaze controller, as we did for instance in [19].

C. Motor babbling

The robot learns the arm-gaze forward kinematics $\mathbf{q}_{gaze} = f(\mathbf{q}_{arm})$ online using LWPR, during a motor babbling phase. The arm motor space has been uniformly sampled to create a set of 20000 target arm configurations; for each of them the robot first moves the arm, and then it gazes at the hand as soon as the arm movement is finished. If the fixation of the hand is achieved, a sample $\langle \mathbf{q}_{arm}, \mathbf{q}_{gaze} \rangle$ is used to train the LWPR. During this process 4400 samples were collected and used for training (the other 15600 arm movements brought the hand to points that could not be fixated, due to joints limits). Other 700 samples were collected moving the arm to random positions, to be used as test set. The estimation error (RMSE, Root Mean Square Error, where the Error is in our case the difference between the real and estimated \mathbf{q}_{gaze}) with respect to the test set was computed during the online update of the LWPR. The trend of the error as the number of training samples increases is shown in Figure 2: the final RMSE is 0.84° .

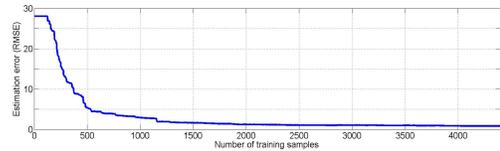


Fig. 2. Estimation error (RMSE) of the arm-gaze forward kinematic model with increasing number of training samples.

The gathered training samples identify points in space that are reachable (i.e. a feasible arm configuration exists which brings the hand to that point). These points are displayed in Figure 3 with respect to different reference frames: Cartesian and gaze-centered. The first two images from the top show the points in Cartesian coordinates (view from the top and view from the side). The highlighted rectangular regions indicate the area in which the reachability test described in Section V is performed. The robot arm and head are also sketched. In the bottom image the same points are plotted in the gaze-centered reference frame, projected on the yaw/vergence plane. Some representative points (A, B, C, D) are highlighted to show the relation between the different representations. In particular, C and D are respectively the farthest and the closest to the robot head; in fact, C is fixated with the minimum vergence ($\theta_v = 11.17^\circ$) and can be reached with the arm fully extended (minimum elbow angle, $\theta_e = 15^\circ$), while D is fixated with the maximum vergence ($\theta_v = 89.15^\circ$) and can be reached with the arm fully bent (maximum elbow angle, $\theta_e = 105^\circ$).

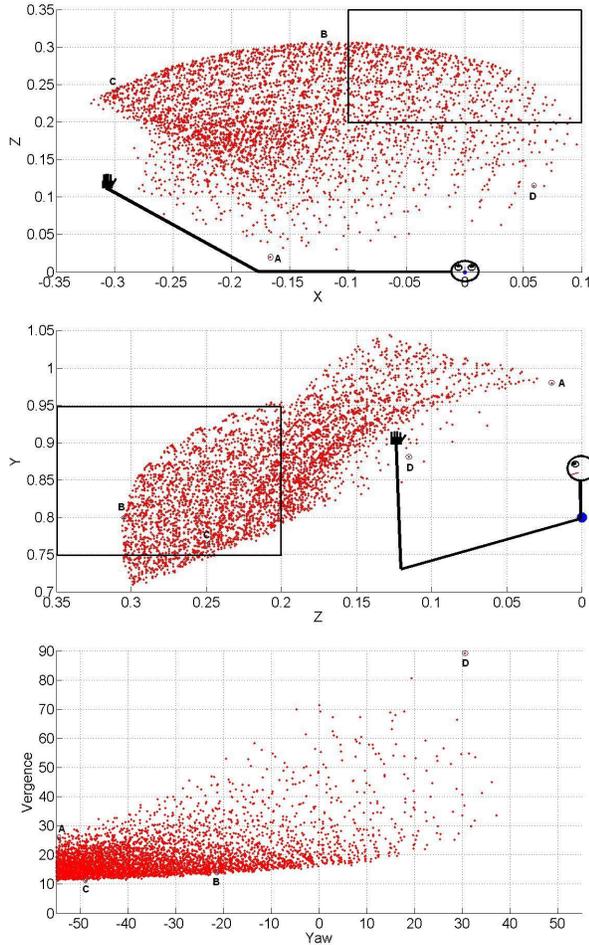


Fig. 3. Reachable points represented in different reference frames and projected on selected dimensions. From the top: Cartesian X and Z (i.e. view from the top), Cartesian Y and Z (i.e. view from the side). The highlighted rectangular regions indicate the area in which the reachability test is performed (see Section V). Then, in gaze-centered reference frame, projected on the yaw/vergence plane. Some representative points (A, B, C, D) are highlighted to show the relation between the different representations.

D. Inverse kinematics

To know whether a fixated object is reachable or not we need to invert the arm-gaze forward kinematics, which means answering the following question: does an arm configuration exist which brings the hand to the current fixation point? Or in other terms, does a valid IK solution exist? As our system is redundant ($\mathbf{q}_{gaze} \in \mathbb{R}^3$ while $\mathbf{q}_{arm} \in \mathbb{R}^4$) we can have also multiple solutions (i.e. the same fixation point defined by \mathbf{q}_{gaze} can be reached with different arm configurations \mathbf{q}_{arm}).

We solve the IK as an optimization problem by using IpOpt (Interior Point Optimizer [28]), a minimization algorithm which has been proven to be fast and reliable (see for example [29] for a precise evaluation of its performance in solving an inversion problem very similar to the one discussed here). The IK computation involves two stages: first we look for a valid solution, then, if at least one exists, we look for an optimal solution according to a given criteria

(thus resolving the redundancy).

The first stage is formalized as follows:

$$\mathbf{q}_{arm}^S = \underset{\mathbf{q}_{arm} \in \Omega}{\operatorname{argmin}} \|\mathbf{q}_{gaze} - \operatorname{fwdKin}(\mathbf{q}_{arm})\| \quad (\text{IV.2})$$

where $\Omega \equiv [\mathbf{q}_{arm}^{min}, \mathbf{q}_{arm}^{max}]$ (i.e. arm joint limits) and \mathbf{q}_{arm}^S is the IK solution. We say that \mathbf{q}_{arm}^S is a valid solution if $\|\mathbf{q}_{gaze} - \operatorname{fwdKin}(\mathbf{q}_{arm}^S)\| = 0$. In that case, the optimal solution is found solving the following problem:

$$\mathbf{q}_{arm}^S = \underset{\mathbf{q}_{arm} \in \Omega}{\operatorname{argmin}} M(\mathbf{q}_{arm}) \quad (\text{IV.3})$$

$$s.t. \quad \mathbf{0} \leq \|\mathbf{q}_{gaze} - \operatorname{fwdKin}(\mathbf{q}_{arm})\| \leq \epsilon \quad (\text{IV.4})$$

where $M(\mathbf{q}_{arm})$ is the measure we want to minimize and ϵ is an arbitrary low error threshold (we set $\epsilon = 0.0001$). We consider a solution optimal if it maximizes the distance of the arm from joints limits, as proposed in [30], and therefore we define $M(\mathbf{q}_{arm})$ as follows:

$$M(\mathbf{q}_{arm}) = \frac{1}{N} \sum_{i=1}^N \left(\frac{\mathbf{q}_{arm}(i) - \mathbf{a}_i}{\mathbf{a}_i - \mathbf{q}_{arm}^{max}(i)} \right)^2 \quad (\text{IV.5})$$

where $\mathbf{a}_i = \frac{\mathbf{q}_{arm}^{max}(i) + \mathbf{q}_{arm}^{min}(i)}{2}$.

E. Training the maps

To produce training samples for the reachable space maps we sampled the gaze motor space uniformly creating a set of 52000 gaze configurations; then we computed IK for each gaze configuration, and stored the results in the form $\langle \mathbf{q}_{gaze}, S, R \rangle$. The value S indicates the presence or not of a valid IK solution (after the first stage of the IK computation): $S = 0.75$ if the solution exists, $S = 0.25$ if it does not. The value R is in inverse proportion to the IK error if no valid IK solution exists (case A), while it is proportional to the optimality if a valid IK solution exists (case B):

$$A) R = \frac{1 - IK_ERR}{2 * MAX_IK_ERR} \quad (\text{IV.6})$$

$$B) R = \frac{1 + OPT}{2} \quad (\text{IV.7})$$

where $IK_ERR = \|\mathbf{q}_{gaze} - \operatorname{fwdKin}(\mathbf{q}_{arm}^S)\|$ and MAX_IK_ERR are the final and maximum IK error after the first stage of the IK computation and $OPT = (1 - M(\mathbf{q}_{arm}^S))$ is the optimality measure after the second stage of the IK computation. R ranges from 0.0 to 1.0, where 0.0 means far from being reachable (high IK error) and 1.0 means reachable with an optimal arm configuration (in our implementation, as far as possible from the arm joints limits). The Basic Map is trained with samples $\langle \mathbf{q}_{gaze}, S \rangle$ while the Enhanced map is trained with samples $\langle \mathbf{q}_{gaze}, R \rangle$.

Since we are interested in building a compact representation that can be queried fast in real-time, we set the LWPR learning parameters in order to have a total number of receptive fields around 10000, and therefore a response time under $2ms$ for a query. We believe that this is a reasonable trade-off between complexity and performances, since the maps are fast enough to be queried in real-time and we were able to achieve good results in terms of accuracy (as we will show in the next Section).

V. EXPERIMENTAL RESULTS

We evaluate here the learning of the reachable space maps. For both maps we show the same set of plots. First we display a 3D visualization of the map training data and of the map output in the gaze configuration space (Figures 4 and 7). Then we show a 2D visualization: a slice on the yaw/vergence plane with pitch = 0° (Figures 5 and 8). With the latter visualization we show the map output not only in relation with the training data, but also in relation with some of the reachable points that were retrieved during the motor babbling phase (in particular, the points that are fixated with pitch $\in [-3.0^\circ, 3.0^\circ]$). Finally, we show the results obtained during a test phase in which the robot gazes at objects positioned at different locations, both reachable and not reachable (Figures 6 and 9). About 9000 locations were obtained by sampling the Cartesian space uniformly within the following ranges: $X \in [-0.10; 0.10]$, $Y \in [0.75; 0.95]$, $Z \in [0.20; 0.35]$ (highlighted regions in the top images of Figure 3). We selected on purpose an area in which reachability borders are present, as those are the more challenging locations for the reachability judgment. Among these positions, the robot was able to fixate about 8000 of them (because of joints limits, the robot could not gaze at some of them): for each of those the output of the maps was computed (in the gaze configuration space) and plotted in the Cartesian space. Projections of the plots on the X/Z and Z/Y planes are shown. Reachable points identified during the motor babbling phase are also shown for comparison. For both maps the output is a continuous value ranging from 0.0 to 1.0 (from red to blue, in the plots). Outputs coloured in blue indicate that the fixated point is considered as reachable.

A. Basic Map estimation

The purpose of this map is to provide a probabilistic information, hence to answer the question “what is the probability of the fixated object to be reachable?”. A fixated point is judged to be reachable if the map output is over a certain threshold.

Since the map is trained with a binary output (reachable / non-reachable) but it produces a continuous output when simulated, it makes no sense to compute the estimation error as a difference between these two values. On the contrary, we compute the percentage of JE (judgement errors about the reachability), and in particular the percentage of FN (false negatives, the training sample indicates the point as reachable while the map output defines it as non reachable) and FP (false positives, vice versa). While JE depends on the learning settings (e.g. total number of receptive fields) and on the nature of the training data, we can play with the design of the map to favor either a lower percentage of FN or FP. We believe is better to have a lower number of FP. In fact, when the map is used for motion planning, the perceived non-reachability of an object would trigger a robot movement which is intended to change the position of the robot with respect to the object, thus making it more easily reachable (e.g. walking toward the object if it is too far); even in the case of a FN, the generated movement will

probably make the reaching action simpler, while in the case of a FP the robot would try to reach for the non-reachable object, and is doomed to fail. For the Basic Map we achieved $JE = 1.3\%$, $FN = 4.7\%$ and $FP = 0.4\%$.

Figure 4 and 5 show visually how the map output matches the training data and how the area defined as reachable (blue area) contains the reachable points extracted during the motor babbling. The results of the test phase are displayed in Figure 6: again, the blue area follows the profile of the reachable points. In the Z/Y plane projection, the white area on the bottom right corner of the plots identifies a region of the Cartesian space that cannot be fixated, due to joints limits.

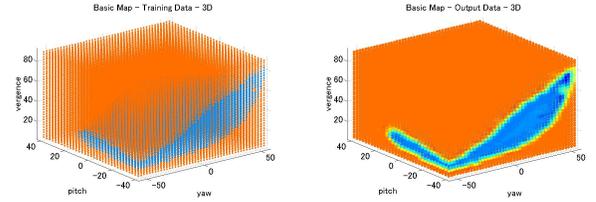


Fig. 4. Basic Map. On the left the training data, on the right the map output after training.

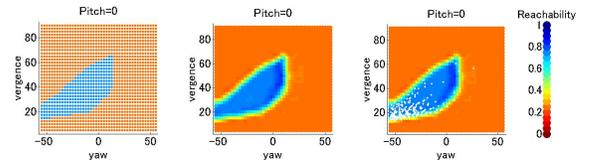


Fig. 5. Basic Map. Projection on the yaw/vergence plane, with pitch = 0°. From the left: training data, map output after training, map output in relation with reachable points (white dots). On the right, the color scale indicating the reachability.

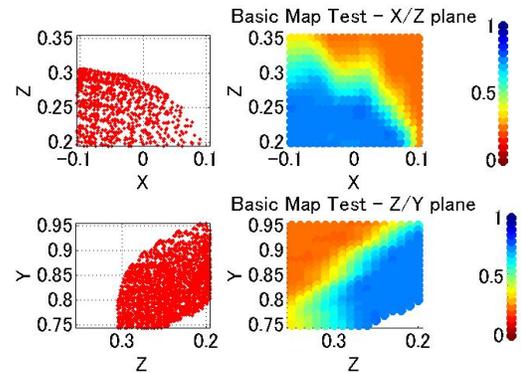


Fig. 6. Basic Map. Results of the reachability test displayed in the Cartesian space, projected on the X/Z plane (top images) and on the Z/Y plane (bottom images). From left to right: reachable points, map output, color scale.

B. Enhanced Map estimation

The purpose of this map is to provide information about the degree of reachability of a fixated point, hence to answer the question “how much / how well is the fixated object reachable?”. The map outputs a continuous value ranging

from 0.0 to 1.0, where 0.0 means far from being reachable and 1.0 means optimally reachable (where the optimality is measured as distance of the arm joints from the limits). The percentage of judgment errors, false negative and false positive is approximately the same as for the Basic Map: $JE = 1.2\%$, $FN = 4.5\%$ and $FP = 0.4\%$. Differently from the Basic Map, here also the training data are continuous and the outputs should be as close as possible to them. Therefore, we can compute the estimation error as the difference between map outputs and training data: the RMSE is equal to 0.022, which means an average error of 2.2%. This small amount of error is evident from the plots in Figures 7 and 8, where the map output strongly resembles the training data. The accuracy of the estimation is also suggested by the plots in Figure 9. Noticeably, the accuracy is not only in judging whether the fixated point is reachable or not, but also in judging the degree of reachability; moreover, the output changes smoothly over the whole space. This aspects make this map the best suited for motion planning, as we will better explain in the next Section.

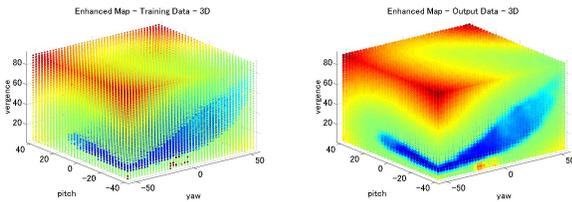


Fig. 7. Enhanced Map. On the left the training data, on the right the map output after training.

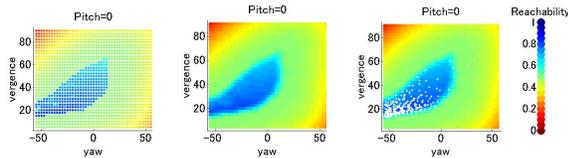


Fig. 8. Enhanced Map. Projections with different values of head pitch orientation. From the left: training data, map output after training, map output in relation with reachable points (white dots). On the right, the color scale indicating the reachability.

VI. POSSIBLE APPLICATIONS

Humans can exploit a variety of movements to extend their possibility to reach for objects. Moreover, their ability to use grasped objects depends on the position of their arm in the workspace: the best performance is achieved only in a relatively small subspace. Similarly, the workspace of a humanoid robot can be huge, depending on its motor skills: to grasp a detected object the robot can decide to rotate or to bend the waist, to bend its knees or to walk toward the object. Furthermore, if the robot places itself optimally with respect to the object, reaching can be achieved with an optimal final arm configuration, easing the subsequent use of the grasped object. The Reachable Space Map we introduced can be used to realize these behaviors. Assuming that the control of gaze

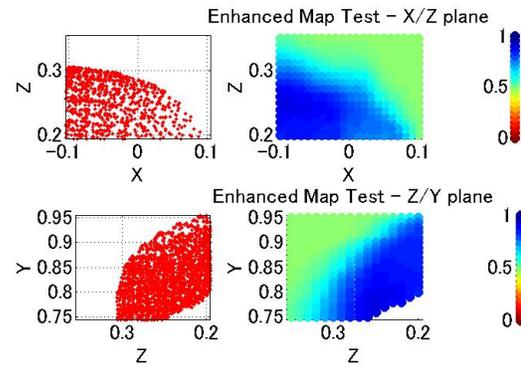


Fig. 9. Enhanced Map. Results of the reachability test displayed in the Cartesian space, projected on the X/Z plane (top images) and on the Z/Y plane (bottom images). From left to right: reachable points, map output, color scale.

is always active, any motion of the robot body with respect to the target object causes the gaze configuration to change along a specific trajectory to keep the object in fixation. The motor-motor relation between the robot motions and the gaze trajectories can be learned from motor experience, for example during an exploration phase in which the robot performs different body motions while keeping an object in fixation. When such motor-motor relations (i.e. kinematic models) are learned they can be used for motor planning: the robot plans a trajectory of the gaze within the Reachable Space Map, in order to change the reachability of an object from the current one to a desired one (e.g. from non-reachable to reachable), and retrieves the body motion that realizes that trajectory. In this respect, it is particularly beneficial that the output of the map is continuous: this allows to use better mathematical tools both for learning the motor-motor relations and for planning and executing the movements.

This application scenario clarifies the advantages provided by the Enhanced Map with respect to the Basic Map. In fact, the robot can plan a body movement which makes a fixated object not only reachable, but also reachable with an optimal arm configuration. In our current implementation we choose the distance from joints limits as optimality measure, but other ones can be used, as for example the distance from singular configurations [31], the *manipulability* measure proposed in [32] or the *reachability index* developed in [23]. Furthermore, the smoother output provided by the Enhanced Map facilitates the generation of gaze trajectories for motion planning, for example using a gradient descent approach; that would be more difficult on the Basic Map, as in some areas a lot of points show very similar outputs (especially inside the non-reachable areas).

A further extension is the creation of a Bimanual Map that join together the maps of both arms: using such a map the robot can position itself optimally in order to perform bimanual operations.

We are currently implementing the complete system on our walking humanoid robot Kobian [1] in order to show the integrated whole-body reaching and locomotion behavior.

VII. CONCLUSIONS

We presented a bio-inspired design for representing the reachable space of a humanoid robot: a Reachable Space Map. The map is encoded in a gaze-centered reference frame, and is learned autonomously by the robot from motor experience. Our solution is innovative with respect to previous works in the choice of both the reference frame and the learning algorithm. As we do not rely on any model of the robot kinematics, our approach is very general and can be applied to any robot equipped with arm and binocular head. We showed results obtained on the dynamical simulator of the humanoid robot iCub, concerning the autonomous generation of two different kinds of maps: the Basic map and the Enhanced map. Both maps provide fast and accurate response when queried. The first one shows a simpler design and can be used to estimate whether a fixated object is reachable or not. The second one encapsulates a richer information about the degree of reachability of fixated objects and seems the best suited for planning more complex robot behaviors such as whole-body and bimanual reaching.

VIII. ACKNOWLEDGMENT

This work was partially supported by RoboSoM project from the European FP7 program (Grant agreement No.: 248366).

REFERENCES

- [1] N. Endo, K. Endo, M. Zecca, and A. Takanishi, "Modular design of emotion expression humanoid robot kobian," in *18th CISM-IFTOMM Symposium on Robot Design, Dynamics, and Control (ROMANSY)*, Udine, Italy, 2010.
- [2] G. Sandini, G. Metta, and D. Vernon, "The icub cognitive humanoid robot: An open-system research platform for enactive cognition," in *Lecture Notes in Computer Science*. Springer Berlin / Heidelberg, 2007, vol. 4850, pp. 358–369.
- [3] C. Galletti, P. Fattori, D. F. Kutz, and P. P. Battaglini, "Arm movement-related neurons in the visual area v6a of the macaque superior parietal lobule," *The European Journal of Neuroscience*, no. 9(2), pp. 410–413, 1997.
- [4] A. P. Batista, C. A. Buneo, L. H. Snyder, and R. A. Andersen, "Reach plans in eye-centered coordinates," *Science*, no. 285, pp. 257–260, 1999.
- [5] J. R. Duhamel, C. L. Colby, and M. E. Goldberg, "The updating of the representation of visual space in parietal cortex by intended eye movements," *Science*, no. 255, pp. 90–92, 1992.
- [6] H. Scherberger, M. A. Goodale, and R. A. Andersen, "Target selection for reaching and saccades share a similar behavioral reference frame in the macaque," *Journal of Neurophysiology*, no. 89, pp. 1456–1466, 2003.
- [7] S. M. Beurze, I. Toni, L. Pisella, and W. P. Medendorp, "Reference frames for reach planning in human parietofrontal cortex," *Journal of Neurophysiology*, no. 104, pp. 1736–1745, 2010.
- [8] W. P. Medendorp, H. C. Goltz, T. Vilis, and J. D. Crawford, "Gaze-centered updating of visual space in human parietal cortex," *The Journal of Neuroscience*, no. 23(15), pp. 6209–6214, 2003.
- [9] J. F. Soechting and M. Flanders, "Sensorimotor representations for pointing to targets in three-dimensional space," *Journal of Neurophysiology*, vol. 62, pp. 582–594, 1989.
- [10] J. McIntyre, F. Stratta, and F. Lacquaniti, "Viewer-centered frame of reference for pointing to memorized targets in three-dimensional space," *Journal of Neurophysiology*, vol. 62, pp. 582–594, 1997.
- [11] D. Y. P. Henriques, E. M. Klier, M. A. Smith, D. Lowy, and J. D. Crawford, "Gaze-centered remapping of remembered visual space in an open-loop pointing task," *The Journal of Neuroscience*, no. 18(4), pp. 1583–1594, 1998.
- [12] A. Blangero, Y. Rossetti, J. Honor, and L. Pisella, "Influence of gaze direction on pointing to unseen proprioceptive targets," *Advances in Cognitive Psychology*, no. 1(1), pp. 9–16, 2005.
- [13] K. Fiehler, I. Schtz, and D. Y. P. Henriques, "Gaze-centered spatial updating of reach targets across different memory delays," *Vision Research*, no. 51, pp. 890–897, 2011.
- [14] M. Flanders, L. Daghestani, and A. Berthoz, "Reaching beyond reach," *Experimental Brain Research*, vol. 126, no. 1, pp. 19–30, 1999.
- [15] Y. Coello, A. Bartolo, B. Amiti, H. Devanne, E. Houdayer, and P. Derambure, "Perceiving what is reachable depends on motor representations: Evidence from a transcranial magnetic stimulation study," *PLoS One*, no. 3:e2862, 2008.
- [16] P. Cardellicchio, C. Sinigaglia, and M. Costantini, "The space of affordances: a tms study," *Neuropsychologia*, vol. 49 (5), pp. 1369–1372, 2011.
- [17] G. Metta, G. Sandini, and J. Konczak, "A developmental approach to visually-guided reaching in artificial systems," *Neural Networks*, vol. 12, no. 10, pp. 1413–1427, 1999.
- [18] C. Gaskett and G. Cheng, "Online learning of a motor map for humanoid robot reaching," in *International Conference on Computational Intelligence, Robotics and Autonomous Systems (CIRAS)*, Singapore, December 2003.
- [19] L. Natale, F. Nori, G. Metta, and G. Sandini, "Learning precise 3d reaching in a humanoid robot," in *International Conference of Development and Learning*. London, UK: IEEE, 2007.
- [20] M. Hulse, S. McBride, J. Law, and M. Lee, "Integration of active vision and reaching from a developmental robotics perspective," *Transactions on Autonomous Mental Development*, no. 2(4), pp. 355–367, 2010.
- [21] E. Chinellato, M. Antonelli, B. J. Grzyb, and A. P. del Pobil, "Implicit sensorimotor mapping of the peripersonal space by gazing and reaching," *Transactions on Autonomous Mental Development*, no. 2(4), pp. 43–53, 2010.
- [22] Y. Guan, K. Yokoi, and X. Zhang, "Numerical methods for reachable space generation of humanoid robots," *The International Journal of Robotics Research*, no. 27, pp. 935–950, 2008.
- [23] F. Zacharias, C. Borst, and G. Hirzinger, "Capturing robot workspace structure: representing robot capabilities," in *International Conference on Intelligent Robots and Systems (IROS)*, 2007, pp. 3229–3236.
- [24] —, "Online generation of reachable grasps for dexterous manipulation using a representation of the reachable workspace," in *International Conference on Advanced Robotics*, 2009.
- [25] N. Vahrenkamp, D. Berenson, T. Asfour, J. Kuffner, and R. Dillmann, "Humanoid motion planning for dual-arm manipulation and re-grasping tasks," in *International Conference on Intelligent Robots and Systems (IROS)*, 2009.
- [26] S. Vijayakumar and S. Schaal, "Locally weighted projection regression: An o(n) algorithm for incremental real time learning in high dimensional space," in *International Conference on Machine Learning (ICML)*, 2000, pp. 1079–1086.
- [27] V. Tikhonoff, P. Fitzpatrick, G. Metta, L. Natale, F. Nori, and A. Cangelosi, "An open source simulator for cognitive robotics research: The prototype of the icub humanoid robot simulator," in *Workshop on Performance Metrics for Intelligent Systems*, National Institute of Standards and Technology, Washington DC, August 19-21 2008.
- [28] A. Wachter and L. T. Biegler, "On the implementation of a primal-dual interior point filter line search algorithm for large-scale nonlinear programming," *Mathematical Programming*, vol. 106, no. 1, pp. 25–57, 2006.
- [29] U. Pattacini, F. Nori, L. Natale, G. Metta, and G. Sandini, "An experimental evaluation of a novel minimum-jerk cartesian controller for humanoid robots," in *International Conference on Intelligent Robots and Systems (IROS)*, Taipei, Taiwan, 2010.
- [30] A. Ligeois, "Automatic supervisory control of the configuration and behavior of multibody mechanisms," *Transactions on System, Man and Cybernetics*, no. 7, pp. 868–871, 1977.
- [31] C. A. Klein and B. E. Blaho, "Dexterity measures for the design and control of kinematically redundant manipulators," *International Journal of Robotics Research*, no. 6, pp. 72–83, 1987.
- [32] T. Yoshikawa, *Foundations of Robotics: Analysis and Control*. MIT Press, 1990.