Abstract—Understanding the mechanism mediating the change from inaccurate pre-reaching to accurate reaching in infants may confer advantage from both a robotic and biological research perspective. In this work, we present a biologically meaningful learning scheme applied to the coordination between reach and gaze within a robotic structure. The system is model-free and does not utilize a global reference system. The integration of reach and gaze emerges from the learned cross-modal mapping between reach and vision space as it occurs during the robot-environment interaction. The scheme showed high learning speed and plasticity compared with other approaches due to the low level of training data required. We discuss our findings with respect to biological plausibility and from an engineering perspective, with emphasis on autonomous learning as well as strategies for the selection of new training data.

I. INTRODUCTION

The transition from inaccurate pre-reaching to accurate reaching observed in infants [15] is one example of the phenomenon of cross-modal mapping whereby links are established between different coordinate systems (e.g. between hand and eye position). Investigating the mechanism underlying this mapping process may confer advantage from a robotic and biological perspective in that it may provide a highly efficient (low learning cycle) method of initial mapping between modalities but biologically, it may also identify a putative developmental mechanism that explains how this phenomenon takes place. This study assesses what is currently known about the biological mechanism, using coordination between hand and eye position as the example, and from that derives a robotic model which is subsequently validated.

Adult human reaching via visual object localization can be performed in a number of different contexts; phasic visual stimuli associated with the object or the object itself can elicit a decision to reach, or reach can be the result of a non stimulus-based cognitive decision whereby the object is either static or dynamic and is searched for. The initial mechanisms underlying each of these scenarios differ in that phasic stimuli immediately orientate a subsequent saccade and foveation on to the object as a reflexive-like activity, whereas saccades during visual finding of a static or dynamic object are internally directed as part of the search decision making process [5], [8]. However, in both situations, movement to the target requires oculo-motor coordination i.e. a transformation of information about the location of the peripheral stimulus or the point of search into specific motor efferents to move the eye accurately to that point. The programming of this transformation to create an accurate direction and distance appropriate movement appears to take place both pre- and post-natally with biological research to date suggesting that there is some initial reflexive directional specification at birth [9] but that this is immediately followed by a calibration process to increase the accuracy of saccade [1], [12]. More recent modeling data from a robotic system provides strong support for this hypothesis [2]. From a developmental perspective, early learning to reach is also associated with error of target location. This occurs up to 3-5 months of age and is often referred to as prereaching [15]. Thereafter, it is considered that infants have a unified coding system within which visual, auditory and proprioceptive stimulation is integrated to facilitate the reach process [3]. What appears to be important about this transition from inaccurate to accurate reaching is that it is not dependent on visual guidance and that after visual or even auditory location of the target, proprioceptive information about hand position is sufficient to attain an accurate reaching action [3]. Research by Thelen et al. (see [13] for review) suggests that this initial mapping by infants occurs through a trial and error process where a wide range of movement parameters and solutions in different contexts and modalities are explored over time in order to calibrate reach movements. For example, mapping between modalities could occur when 1) an object is touched and then saccaded to, 2) when an object is located through visual or auditory stimuli and then touched or 3) when an object is placed and then saccaded to.

Biologically, a calibration process that could account for all of these different modalities and situations is still unknown. One possible method is a simple mapping strategy that links the location of the object identified through one modality with another. This is referred to here as a learning scheme for cross-modal mapping and the aim of this study was to examine this strategy (using the aforementioned example 3 [known proprioceptive location through object placement mapped to eye-coordinates through saccade]), from the perspective of robotic efficiency (rate of learning) but also to critically assess it as a putative biological mechanism underlying the phenomenon of cross-modal mapping.

II. ROBOT SYSTEMS, CONTROL AND LEARNING TASK

The robot hand-arm system, the active vision system and their spatial organization on and around a table can be seen in Fig. 1. Mounted on the table, the purpose of the arm is to pick up an object as well as to put a grasped object down at a pre-defined location on the table. In this setup we always deal with
A. Vision and hand-arm system

The active vision system integrates two cameras (both provide RGB image data, 1032x778, 25 frames per second) mounted on a pan-tilt-verge unit. In this experiment we didn’t use the pan movement. Hence, the active vision system has 3 degrees of freedom (DOF), that is, one verge movement for each camera and one tilt which moves both cameras. Each motor can be controlled by determining the values for speed or position, given in radians (rad).

The robot arm and hand systems (manufacturer SCHUNK GmbH & Co. KG) have 7 DOF each. We make use of only five DOF of the arm in order to place the robot hand at certain positions on the table. The hand system has three fingers. Each finger has two segments each equipped with a pressure sensitive sensor pad. Since the control of the grasping is out of the scope of this paper we won’t give any further details about the hand system and its control.

B. Reaching and gaze-control

The domain of the reach movement, referred to here as reach space, is represented as a 2-dimensional domain because the objects are only located on a table, a 2-dimensional space. Taking the base of the arm as reference, a table location is fully determined by the distance \(d\) (cm) and the planar angle of the arm \(\alpha\) (rad) (see Fig. 1). The inverse kinematics mapping the the 2-dimensional reach space to the 5-dimensional joint space of the arm is solved analytically which won’t be described further. It is important to note that arm-control only places the hand on the table with respect to a given distance and orientation \((d, \alpha)\).

The purpose of the gaze-control is to move the cameras in such a way that the visual stimuli, the colored ball, will be driven into the image center. To achieve reasonable performance RGB image data of resolution 1032x500 were captured and reduced to 129x62. Each RGB color value in the reduced image represents the mean value of the color values in the corresponding 8x8 sub-field in the original image. The reduced RGB images were filtered with respect to a defined color, here blue. Non-zero values (grey pixels) in the filtered data indicate the appearance of the filtered color in the image. The \(x\) and \(y\) positions of the non-zero values, actually their distance from the image center \((x_c, y_c)\), relate to specific speed values \(s_x\) and \(s_y\). These values are used to specify the speed values of the motors controlling verge and tilt (Fig. 2). The relation between \(x\) and \(y\) and speed values is linear:

\[
\begin{align*}
    s_x &= \frac{2 \cdot (x - x_c)}{2x_c}, \\
    s_y &= \frac{2 \cdot (y - y_c)}{2y_c}
\end{align*}
\]

where \(2 \cdot x_c\) and \(2 \cdot y_c\) is the horizontal and vertical resolution of filtered image, respectively. The actual speed values of the verge \(v_x\) and tilt \(v_y\) motors are calculated as follows:

\[
\begin{align*}
    v_x &= c_x \cdot \overline{s_x}, \\
    v_y &= c_y \cdot \overline{s_y}
\end{align*}
\]

where \(c_x \cdot v\) are constants for normalization while \(\overline{s_x}\) and \(\overline{s_y}\) represent the mean values of all non-zero values of \(s_x\) and \(s_y\) values in the color filtered image data, respectively.

In order to avoid conflicts between different tilt values resulting from the different visual input of left and right cameras, only the left camera controls two DOF: its verge and the tilt. The right camera can only determine its own verge movement. In consequence, the right camera might not be able to drive the visual stimuli completely into its center.

The signals coming from the gaze control drive the motors of the tilt and verge axis directly. However, if the stimulus is shifted into the image center then the motor signals become zero and the system comes to a standstill indicating that it has focused on the stimulus. Such a halt position is fully determined by the motor positions of the tilt, left and right verge axis, \((p_{tilt}, p_{L, verge}, p_{R, verge})\), we call this parameter space the vision space.

C. Overall system architecture and learning process

Fig. 3 illustrates the general system architecture which combines the arm and the active vision systems. Both systems act independently and can be seen as separated sensorimotor systems. A coordination between them is achieved by the
central unit. In the case of the arm system, the central unit can set target coordinates \((d, \alpha)\) in the reach space which triggers specific hand movements as well as requesting state information, for example the current hand states or whether the target position has been reached. Regarding the active vision system the central unit switches on and off the gaze control and can read and set positions of the motors driving the tilt, left and right verge axis, \((p_{\text{tilt}}, p_{\text{vL}}, p_{\text{vR}})\).

Bridging the arm and vision system, the central unit drives the learning of the hand-eye coordination by learning the mapping between reach space and vision space resulting from the interaction of vision and arm system in its shared environment, the table. It is essential for the arm system to know its current position (i.e. proprioception) which leads to a specific position of the ball on the table. There has to be some token of location that can be related to the eye contingency. The physical arm/vision configuration in space does not have an impact on the algorithm. Any configuration is possible that allows the vision system to “see” the table completely.

Learning the cross-modal mapping between reach and vision space is done by a kind of case-based learning strategy. The mapping, we have implemented, stores the pairs \([(d, \alpha), (p_{\text{tilt}}, p_{\text{vL}}, p_{\text{vR}})]\) representing concrete examples of bidirectional links between reach and vision space. With such a mapping, the coordination of reach and gaze is simply achieved by deriving the \((d, \alpha)\) location in reach space when the active-vision system configuration \((p_{\text{tilt}}, p_{\text{vL}}, p_{\text{vR}})\), resulting from the gaze-control, is given. Conversely, if table position \((d, \alpha)\) is given then the mapping delivers directly the tilt-verge positions which drives the active vision system to look where the arm has reached to. Generalization is achieved by introducing a metric which allows the definition of a distance measure between two points in the same space, reach or vision. Here we applied the Euclidean distance in the vision space. A distance measure is necessary since the stored example configurations are very unlikely to occur again. In fact, we only search for the closest neighbor stored in the mapping. This closest neighbor leads to the best estimation of the corresponding point the mapping can provide. This learning scheme is inspired by a previous methodology [11] used to learn the sensorimotor mapping for saccadic eye movements in a robot system.

### III. Experiments

In order to enable the system to learn or to build up the mapping between reach and vision space autonomously we have implemented a protocol (large box) that runs without human intervention. The arm system “presents” to the vision system an object at a known position, initiates the gaze control and the central unit stores the resulting configurations in the mapping structure, represented as link between a point in vision and a point the reach space.

According to this protocol the system just collects concrete examples how an object location relates to a specific tilt-verge configuration. An evaluation of the developing mapping with respect to the precision of estimating an object’s location for a given tilt-verge configuration is done in a second test procedure (small box).

The protocol for testing a given mapping is similar to the learning procedure. It only differs in step 4 and the initialization of the mapping in step 1. Here we start with a learned mapping which is not subject of any adaptation during the test. Instead, on the base of the given mapping \(M\) and the configuration of the vision system after the gaze \((p_{\text{tilt}}, p_{\text{vL}}, p_{\text{vR}})\) an estimation of the table position \((d_e, \alpha_e)\) is derived. The distance between estimated \((d_e, \alpha_e)\) and actual

---

**Protocol for learning the mapping:**

1. **Initialize empty mapping** \(M\)
2. **Pick up object**
   2.1 - move arm to pre-defined object position
   2.2 - pick up object
   2.3 - move arm away
3. **Select a target positions in reach space** \((d, \alpha)\)
4. **Execution of reach and gaze**
   4.1 - move arm to the target position
   4.2 - put object on the table
   4.3 - move arm away
   4.4 - start gaze control
   4.5 - wait until vision system has the object in its focus
   4.5 - stop gaze control
5. **Learning**
   5.1 - read position values \((p_{\text{tilt}}, p_{\text{vL}}, p_{\text{vR}})\)
   5.2 - add example \([([d, \alpha], (p_{\text{tilt}}, p_{\text{vL}}, p_{\text{vR}})])\) to \(M\)
6. **Pick up the object**
   6.1 - move arm to the target position
   6.2 - pick up object
   6.3 - move arm away
7. **Go back to 3**
Protocol for testing a given mapping:

1. Initialize given mapping $\mathcal{M}$
2. Pick up object
3. Select a target position in reach space $(d, \alpha)$
4. Execution of reach and gaze
5. Test
   5.1 - read position values $(p_{\text{tilt}}, p_{\text{cL}}; p_{\text{cR}})$
   5.2 - $(d_e, \alpha_e) := E(\mathcal{M}; p_{\text{tilt}}, p_{\text{cL}}; p_{\text{cR}})$
   5.3 - calculate distance between $(d_e, \alpha_e)$ and $(d, \alpha)$, estimated and actual position
6. Pick up the object
7. Go back to 3

$(d, \alpha)$ table position is applied as a measure of the mapping precision.

In this experimental setup we have investigated the following three issues. First, which level of precision can be achieved by this setup. Second, how many examples are needed to achieve a certain precision. And third, what is the importance of how new examples are selected.

With respect to the last question we compare two strategies for selecting new examples. Since the learning is always initiated by the arm system, the selection of new learning examples is reduced to the question, how to select a new point in the reach space. In the first case the target position on the table is selected randomly. This random selection strategy $S_{\text{random}}$ results in a mapping we refer to as $\mathcal{M}_{\text{random}}$. The other strategy $S_{\text{gaps}}$ goes for the “gaps on the table.” A new randomly selected target table position has to have a minimum distance to all table positions already stored in the map. At the beginning of the learning we start with a certain value of minimum table distance (here 25cm). This value will be divided by two if no gaps of this size can be found in the map anymore. In such a way we achieve an equal coverage of the table space right from the beginning of the learning process. We call the resulting mapping $\mathcal{M}_{\text{gaps}}$.

We have applied both selection strategies in two separated runs. Both mappings, $\mathcal{M}_{\text{gaps}}$ and $\mathcal{M}_{\text{random}}$, contain 300 links. For the test we have recorded another run of 100 examples. A recording of 100 examples following the described protocols for learning or testing requires approximately 1.5 hour.

In order to address the first two questions about precision and the required number of examples we have investigated the relationship between precision and mapping size, i.e. number of stored examples.

The actual table space the system is operating in is defined by the range of distance $d$ and angle $\alpha$, here we have: $-1.4 \leq \alpha \leq 1.4$ (rad) and $30 \leq 60$ (cm) spanning an area of 3944 cm$^2$.

IV. RESULTS

The overall test of the system is its ability to pick up an object by estimating its position based on the tilt-vergence configuration only. This is the only source of information about the object location if it was placed on the table by an external source, e.g. a human or another robot system. It turned out that both mappings provide sufficient accuracy to reach, grasp and pick up the object.

The way we estimate a table location $(d_e, \alpha_e)$ based on the current configuration $(p_{\text{tilt}}, p_{\text{cL}}; p_{\text{cR}})$ and a given mapping $\mathcal{M}$ is formally written as: $(d_e, \alpha_e) := E(\mathcal{M}; p_{\text{tilt}}, p_{\text{cL}}; p_{\text{cR}})$. All three components $p_{\text{tilt}}$, $p_{\text{cL}}$ and $p_{\text{cR}}$ are used in the “estimation” function $E$.

The diagrams in Fig. 4 summarize the development in estimation performance related to the size of the mapping. In the top, the average values are plotted over the size of the mapping, while on the bottom diagram shows the standard deviation. Each mapping size was tested with the same test set of 100 examples.

The diagrams indicate that the random selection of table positions needs a significantly larger mapping size to achieve the same accuracy compared with the strategy which goes for the gaps in the unexplored reach space. After 300 links the $S_{\text{gaps}}$ has a significantly better average error of 2.0 +/- 1.2 cm compared with 2.7 +/- 1.5 cm of $S_{\text{random}}$. Moreover, the evolution of the average error value over the number of links of $S_{\text{gaps}}$ indicates a saturation at 2.0 cm, meaning better results can’t be expected for this robot system. Furthermore, the data show that the average error drops with the mapping size until it saturates at a certain level of accuracy. This fact shows that the system improves with each new example stored in the mapping structure. Due to noise, these curves aren’t monotonously decreasing in the mathematical sense but the general tendency is.

To get an impression about the data the system has actually learned, we have plotted the points establishing the links between the reach (left, Fig. 5) and vision space (right). For reasons of representation we only have represented the 2-dimensional sub-space of the vision space (defined by tilt,
verage left). The plots in Fig. 5 clearly indicate that selection strategy $S_{gap}$ guarantees significant better coverage of the relevant areas in the vision space. Notice the measure defining distance in reach space is based on the distance on the table, this is different than just using Euclidean distance in the reach space. In fact, although different points in the plots on the left seem to have almost the same distance, meaning the same Euclidean distance, it might be that their corresponding distance on the table in the external world is significantly different.

One can see that the coverage in the vision space is what determines the precision we are aiming for because the estimation performance of a table position depends on the coverage of the vision space. Furthermore, it is interesting to see that good coverage of the table area corresponds to good coverage of the relevant areas of the vision space.

Fig. 5. Points in reach (left) vision space (right) the building links which establish the mapping between the two spaces. Both mappings are learned with different selection strategies, purely random (top) and while going for the gaps (bottom).
up an object on the table. Interestingly, a precision value of 2 cm was set as the minimum requirement for competence, thus, the latter section of the learning phase using the $S_{gap}$ strategy was in fact redundant in this regard. Theoretically, if some form of self-evaluation of competence had been in place, then this redundancy could have been avoided and learning could have stopped much earlier. Such an evaluation, as to whether or not the object was picked up and, therefore, as to whether or not the estimation was good enough, could be provided by the tactile sensors of the fingers. In this respect, it is not hard to imagine the advantage of a system that could combine the learning and testing protocol and thus permanently evaluate its estimation, stopping learning when necessary but also adding new links when a poor estimation is made. In this situation, there would be no difference between the learning and testing phases but rather each test could directly be used as an additional learning example.

D. Constructing space without a global coordinate system

From the perspective of an external observer we can say that the learned hand-eye coordination process has established an area of useable space for the robotic system. This space (or domain) has no global reference system but emerges from the relation between arm-hand and active vision system as it occurs during the robot-environment interaction. In other words, through this learning process the system has constructed a representation of space which is determined by its visual and proprioceptive capabilities.

From an engineering perspective we can point to several advantages of such a construction of space without global coordinate system. Firstly, neither the vision system nor the arm need to be calibrated with respect to a global coordinate system. Secondly, there is also no need to know the exact relative relation between arm and vision system. The system learns the relation between reach and vision space without these details. Finally, due to the total separation of the two sensorimotor systems (arm and vision system) the relationship between vision and reach space can be learned in low dimensional spaces which supports fast learning.

E. Biological plausibility

The attraction of case-based learning from a biological perspective is that it embraces two central concepts within the field of developmental research; firstly Thelen’s idea of exploration and selection [14] and secondly Piaget’s original dogma on schemas whereby both case-based learning and schemas can be described as non-modality specific, context driven opportunities for learning [7]. Although this provides a strong argument for biological plausibility, it cannot be logically extended to infer information about the actual underlying biological mechanism which, due to the non-neural network nature of the model, is likely to be quite different. However, the model may actually suggest greater similarities to the biology since the learning speed and the low number of learning cycles are much closer to biological systems compared to that of neural networks [10].

VI. CONCLUSION

In this work we have introduced a fast learning scheme and demonstrated its performance in a robotic scenario aiming at the integration of reach and gaze-control. This learning scheme provides fast learning and high plasticity. We have shown that a robust cross-modal mapping between reach and vision space can be learned without making use of a global coordinate system and we have outlined that such an approach provides significant advantages in engineering autonomous systems. As we have argued at the beginning, the mapping scheme between vision and reach space as a substrate for hand-eye coordination is backed by investigations in biological systems. On the other hand, our actual implementation of this mapping might be considered as purely engineered and far away from being biological plausible compared with implementations based on artificial neural networks. However, if we consider learning speed, plasticity and continuous autonomous learning then our learning scheme seems to meet some of the essential abilities of learning processes embodied by biological systems.

ACKNOWLEDGMENT

The first two authors gratefully acknowledges the support of EPSRC through grant PEP/C516303/1.

REFERENCES