Sensorimotor coupling via Dynamic Bayesian Networks

Ruben Coen Cagli, Paolo Napoletano, Paolo Coraggio, Giuseppe Boccignone, Agostino De Santis

Abstract— In this paper we consider the problem of sensori-motor coordination in a Bayesian framework. To this end we introduce a novel kind of Dynamic Bayesian Network serving as the core tool to integrate active vision and task-constrained motor behaviors. The proposed system is put into work by addressing the challenging task of realistic drawing performed by a robotic agent, namely a 7-DOF anthropomorphic manipulator. Simulation results are compared to those obtained by eye-tracked human subjects involved in drawing experiments.

I. INTRODUCTION

Any agent, either biological or artificial, situated in the world requires the combined effort of both perceptual (sensory) and action–related (motor) resources in order to survive. However, which are the criteria that guide such critical liaison is still an open issue.

It is evident that both motor and perceptual behaviors of an agent are constrained; in humans, for instance, eye movements are task dependent, as they are targeted to extract the relevant information for task resolution [1]. On the other hand, recent approaches to sensorimotor coordination in primates claim that motor preparation has a direct influence on subsequent eye movements [2], sometimes turning coor-dination into competition. Complementary, eye movements come into play in generating motor plans, as suggested by the existence of *look ahead* fixations in many natural tasks [3]. To investigate the sensorimotor coordination issue we consider the task of realistic drawing. Realistic drawing is not considered as a "common" visuomanual activity such as driving, washing one's hands or making a cup of tea [3], neither a "common" visual task such as the recognition of a face; indeed drawing requires a high precision of hand movements and a high degree of voluntary attentional control in directing fixations both on the scene and on the drawing hand.

A second, but related issue, is that any agent situated within the world has to contend with uncertainty about the world and with noise plaguing its sensory inputs and motor commands; in this perspective, the Bayesian approach provides a powerful framework [4].

To face both issues, here we present a novel kind of Dynamic Bayesian Network (DBN) [5], the *Input–Output*

R. Coen Cagli is with Department of Neuroscience, Albert Ein-stein College of Medicine of Yeshiva University, Bronx NY, USA `rcagli@aecom.yu.edu`

P. Coraggio is with DSF, Università di Napoli "Federico II", Via Cintia, Napoli, Italy `coen,pcoraggio@na.infn.it`

P. Napoletano e G. Boccignone are with the Natural Computation Lab DIIIE, Università di Salerno,via Ponte Don Melillo, 1 Fisciano (SA) Italy `pnapoletano, boccig@unisa.it`

A. De Santis is with the PRISMA Lab DIS, Università di Napoli, Via Claudio 21, Napoli, Italy, `agodesa@unina.it`

Coupled Hidden Markov Model (IOCHMM) which provides a general high level mechanism for the dynamic integration of eye and hand motor plans, and enables the use of information coming from multiple sensory modalities. The model also accounts for the task–dependence of eye and hand plans, by learning a sensorimotor mapping that is suitable for the given task, namely realistic drawing.

Further, we show how the IOCHMM can be suitably employed as the core of a system that allows a robotic arm to produce human–like eye–hand movements in solving a realistic drawing task, where the robot is allowed to look at an image while copying it. The proposed system is outlined at a glance in Fig. 1.

Clearly, drawing, as well as many anthropic tasks, involves force and position constraints for the robotic arm, for what concerns inverse kinematics and trajectory planning. Here, the inverse kinematics is computed using a closed–loop algorithm with redundancy resolution. It is worth noting that, for a human draughtsman, when a picture to be reproduced is observed, the direction of gaze is crucial for trajectory planning, and the postures assumed while executing such task are strictly related to optimizations shaped by evo-lutionary history. When a 7 degrees of freedom (DOF) anthropomorphic robot manipulator performs a drawing task or a handwriting task, the 4 redundant DOF's can be suitably used to perform secondary tasks [7]. In this work the redundancy is handled considering a minimization of joint velocities (use of the Moore–Penrose pseudo–inverse of the Jacobian matrix), and additional tasks can be considered [8].

Simulation results are provided: the outputs of the pro-posed model, namely the scanpath as well as the task–space and joint–space motions of a 7 DOF robot, in the drawing task, are compared to human eye movement recordings and hand movement analysis. Experiments have been performed with 25 human subjects. Preliminary results give evidence of reasonable performance in comparison with human vi-suomotor behavior.

II. JOINT EYE–HAND MOVEMENT PLANNING

The selection of the coordinate sequence of eye and hand actions, namely saccades and hand trajectories, calls for reliable inference (via IOCHMM) and decision of such actions on the basis of visual and proprioceptive inputs.

A. Visual and proprioceptive processing

Consider again the system in Fig. 1. The visual input is represented by the image of the observed world scene while the reafferent proprioceptive input is represented by the velocity of the end effector in the drawing plane.
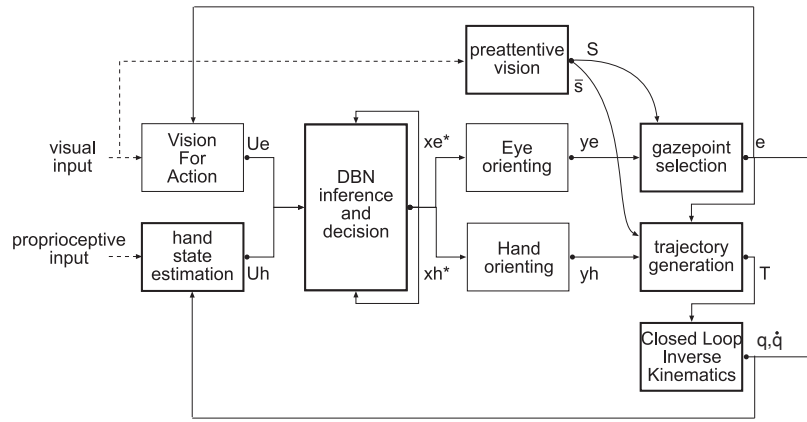
Fig. 1. A system for sensorimotor coordination of a robot involved in a realistic drawing task (see text for explanation)

Clearly the proprioceptive input can be hardly measured in experiments with human subjects, but cortical recordings in behaving non–human primates suggest that populations of neurons can encode 2D hand velocity [20].

The proprioceptive input is fed into the the *Hand State Estimation* module which by taking into account internal feedback computes an estimate of the hand direction u^h (radians), $u^h \in \left\{0, \frac{\pi}{4}, \ldots, \frac{7\pi}{4}\right\}$ while here we are not interested in speed.

The visual input, which more precisely is represented by the scene image together with the point fixated within that image (the fixation **e** provided as visual feedback, cfr. Fig. 1 follows two routes. In the *Preattentive Vision* module, early visual features (color, intensity, orientation) are extracted through linear filtering across different scales; then, center–surround differences are computed for each feature to yield the feature maps, that are combined in the saliency map **S** (see [9] for details). From such map a list $\bar{\sigma}$ of the n ($n \simeq 6, 7$) most salient points in **S**; such points represent bottom-up, plausible FOA candidates that can bias high level gaze planning.

In the *Vision for Action* module, action–related information [10] is computed, within the image region surrounding the fixated point **e** (such region represents the the Focus of Attention, FOA) to provide subsequent modules with orientation and curvature information. More precisely, due to the peculiar characteristics of realistic drawing [11], a regular grid is ideally superimposed on the original image and two matrices (N, O) are obtained by assigning to each cell respectively an on/off intensity value (Fig. 6(d)) and the average orientation of the contour (Fig. 6(e)). Eventually, the visual feature u^e, forwarded to the DBN, is coded as an angular value corresponding to the orientation value of the image contour in the currently fixated cell, and takes the following values (radians), $u^e \in \left\{0, \frac{\pi}{8}, \ldots, \frac{7\pi}{8}\right\}$.

B. Inference of sensorimotor behavior through the IOCHMM net

At this level, appropriate plans for sequences of eye and hand movements are generated, following a Bayesian approach.

The sequential nature of gaze allocation, and the observation that saccades are driven by neural signals that are inherently noisy, suggest that scanpaths are best described by stochastic processes [12], [13]. Furthermore the variability in observable quantities, e.g. fixation duration and saccade length, reflects not only random fluctuations in the system but also factors such as moment–to–moment changes in the visual input, cognitive influences, and the state of the oculomotor system. To account for these further variables, most recent models of eye movements in reading [14] have adopted the Input–Output HMM (IOHMM, see [15]). In the IOHMM the temporal evolution of the (hidden) state variable is described as a Markov process, but conditioned on some observed (or input) variables.

Similar considerations hold for motor planning, and this is the main reason for the widespread diffusion of probabilistic techniques in modeling sensorimotor behaviors in humans and animals [4]. Furthermore, probabilistic graphical models together with Bayesian Decision Theory are a rich tool not only for modeling biological systems (the inverse problem, fitting the data), but also for controlling artificial agents (the direct problem, generating/simulating the data) [4].

Here we introduce a module based on probabilistic network capable to learn the sensorimotor mapping, whose inferences are then evaluated in the framework of Bayesian Decision Theory, to select the best eye–hand movements given the multimodal input and previous movements.

Since dealing with a process unfolding in time, the network proposed is shaped as a DBN [5] for which the graph in Fig. 2 depicts pictures two temporal slices; nodes denote random variables, and arrows, conditional dependencies.

In detail, the process corresponding to the temporal evolution of the eye plan is modeled in our network as an IOHMM; we have three layers of variables, i.e. input variables u^e, u^h related to vision and proprioception respectively, and the hidden and output variables x^e, y^e corresponding to the eye plan. Similar considerations hold for the hand plan, where the inputs are the same, while the hidden and output variables are denoted x^h, y^h.
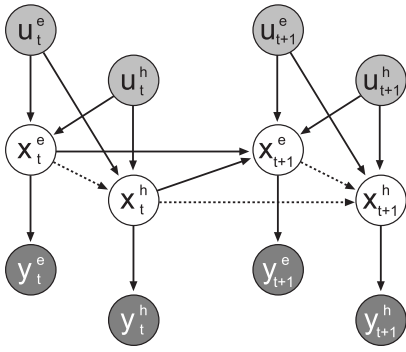
Fig. 2. The IOCHMM's for combined eye and hand movements. The gray circles denote the input (**u**) and output (**y**) variables. Dotted connections in the hidden layer highlight the subgraph that represents the dependence of hand movements on eye movements, while continuous connections denote the dependence of the eye on the hand.

The most important point here is that the two processes should not be considered as independent (see section I), but rather as coupled chains: a viable solution is to consider a graphical model that unifies the IOHMM Model and another DBN known in the literature as the Coupled HMM [5], [16]. We call the resulting DBN an *Input–Output Coupled Hidden Markov Model* (IOCHMM, Fig. 2)

By generalizing the time slice snapshot of Fig. 2 to a time interval $[1, T]$ we can write the joint distribution of the state and output variables, conditioned on the input variables as:

$$
\begin{aligned}
p(\bar{x}_{1:T}, \bar{y}_{1:T} \,|\, \bar{u}_{1:T}) = \\
p(x_1^e \,|\, u_1^e, u_1^h) p(y_1^e \,|\, x_1^e) p(x_1^h \,|\, u_1^e, u_1^h, x_1^e) p(y_1^h \,|\, x_1^h) \cdot \\
\prod_{t=1}^{T-1} \Big[p(x_{t+1}^e \,|\, u_{t+1}^e, u_{t+1}^h, x_t^e, x_t^h) p(y_{t+1}^e \,|\, x_{t+1}^e) \cdot \\
p(x_{t+1}^h \,|\, u_{t+1}^e, u_{t+1}^h, x_{t+1}^e, x_t^h) p(y_{t+1}^h \,|\, x_{t+1}^h) \Big]
\end{aligned} \quad (1)
$$

where $\bar{u}_{1:T}$ denotes the pair of input sequences from $t = 1$ to T $(u_{1:T}^e, u_{1:T}^h)$, $\bar{x}_{1:T}$ denotes the pair of state sequences and $\bar{y}_{1:T}$ the pair of output sequences.

Hidden variables x^e, x^h are assumed to be discrete, taking values in $\left\{ 0, \frac{\pi}{4}, \ldots, \frac{7\pi}{4} \right\}$.

In order to use the DBN as a control system, we apply a decision step to inference results. According to Bayesian Decision theory, if the agent is given a set of observation data d and formulates some hypotheses h, a *decision rule* is a function $\alpha(d)$ that associates the data with an hypothesis. A loss function $L(h, \alpha(d))$ can be defined to quantify the cost of choosing a wrong hypothesis. Then, the agent will take its decision so as to minimize the *risk*, namely a functional that quantifies the cost of the decision weighted by the joint probability of the data and the hypothesis:

$$
r(\alpha) = \Sigma_{h,d} L(h, \alpha(d)) P(h, d) \quad . \quad (2)
$$

Different choices can be made for the cost function, and the decision rule that minimizes the risk changes accordingly. In the simulations presented in this work we used the most

basic cost function:

$$
L(h, \alpha(d)) = \begin{cases} 0 & \text{if } \alpha(d) \neq h \\ 1 & \text{otherwise} \end{cases} \quad . \quad (3)
$$

With this choice the risk function is minimized when the decision rule just selects the hypotheses that maximizes the conditional probability $P(h|d)$. In our case by substituting h with the pair (x_{t+1}^e, x_{t+1}^h), the decision rule selects the eye–hand plans that are the $\arg\max$ of the state transition distribution:

$$
\begin{aligned}
(x_{t+1}^{e\star}, x_{t+1}^{h\star}) = \\
argmax \Big[p(x_{t+1}^e, x_{t+1}^h \,|\, u_{t+1}^e, u_{t+1}^h, x_t^e, x_t^h) \Big] \quad .(4)
\end{aligned}
$$

Note that, after the DBN has been trained with a sufficient number of examples (see section IV), it is straightforward to compute at any time step t, given the inputs (u_t^e, u_t^h), the values of the state transition distribution $p(x_{t+1}^e, x_{t+1}^h \,|\, u_{t+1}^e, u_{t+1}^h, x_t^e, x_t^h)$. Moreover, such posterior probability takes into account a factor that prevents from moving the eye towards empty cells.

Eventually, the values of the outputs $(y_{t+1}^{e\star}, y_{t+1}^{h\star})$ are found by sampling the corresponding output distributions conditioned on the hidden states, and choosing the most likely values, which is performed by the *Gaze orienting* and *Hand orienting* modules, respectively. Here, similarly to related input and hidden variables, $(y_{t+1}^{e\star}, y_{t+1}^{h\star})$ are assumed to be discrete variables, taking values in the set $\left\{ 0, \frac{\pi}{4}, \ldots, \frac{7\pi}{4} \right\}$.

C. Gazepoint selection and hand trajectory generation

Planning of hand trajectory is achieved by fusing the outputs of different sensorimotor modules of our architecture, in the *Trajectory Generator* module (see Fig. 1); here the goal is to reproduce the trajectory planning strategy that can be inferred from the observation of human draughtsmen. Eye tracking experiments have shown that the most common drawing behavior is the following: a) subjects fixate on a location on the original image, b) then move the gazepoint towards the pencil tip, c) draw the corresponding portion of the image and d) stop drawing and go back to point a). Such a cyclic behavior is illustrated in Fig. 5, where it is possible to observe the temporal sequence of drawing movements by the subject whose scanpath is shown in Fig. 4(a). Accordingly, in our model hand trajectory is generated and executed in segments, and the endpoints and intermediate key points of each segment are defined by the fixation points.

Recall that, at any given time step t, the *Gaze orienting* and *Hand orienting* modules provide the next eye and hand movement directions, (y_t^e) and (y_t^h), respectively; meanwhile a set of most salient points $\bar{\sigma}$ is made available by the *Preattentive Vision* module. The latter points are translated to the coordinates of the hand workspace, and are used as the starting and ending points for each trajectory segment.

Gaze points **e** are determined by the *Gazepoint Selection* module as follows. Assume a current gaze location $\mathbf{e}_t \in \bar{\sigma}$; given $\mathbf{e}_t$ and the value of y_t^e, the cell where the gaze point
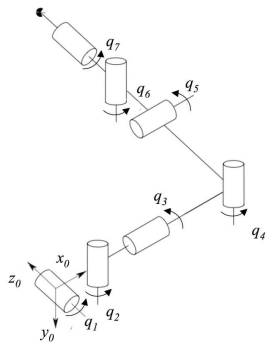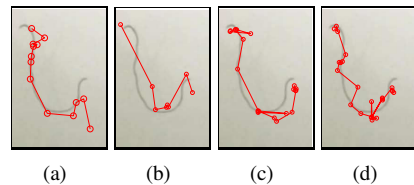
Fig. 3. The 7 DOF manipulator

Fig. 4. The scanpath executed by a human subject in the drawing task. See 5 for the corresponding hand trajectories
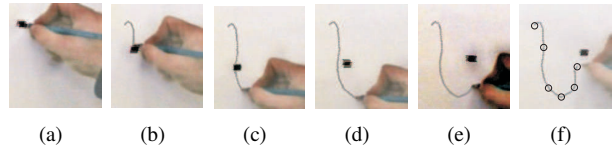

Fig. 5. The sequence of hand movements by a human subject in the drawing task. The solid black square denotes the fixation. In 5(f) the circles denote the endpoints points of each trajectory segment, found by inspection of the video recording, as the points where the drawing movement stops for a while. See Fig. 4(a) for the corresponding scanpath.

will move next (ϵ_{t+1}) is computed. Then, the next gaze location $\mathbf{e}_{t+1}$ is obtained by finding the most salient point in the image patch $I(\epsilon_{t+1})$ corresponding to the next cell. The gaze point $\mathbf{e}_t$ and the angular value ϕ_t of the chosen hand direction y_t^h are fed into the *Trajectory Generator* module. This is repeated until that $\mathbf{e}_{t+\tau} \in \bar{\sigma}$; in this case the sequence of pairs $[(\mathbf{e}_{t+i}, \phi_{t+i})]_{i=0,1,\ldots\tau}$ is interpolated by a spline, setting the slope of the curve at point e_{t+i} to the value $tan(\phi_{t+i})$. The resulting curve is the trajectory segment that is fed into the *Inverse Kinematic* module for generating actual motor commands.

III. INVERSE KINEMATICS

Movements of a redundant seven degree–of–freedom (DOF) robot manipulator, having a human–like kinematic structure (Fig. 3), have been simulated.
The drawing task considered in this paper leads to a solution to the inverse kinematics that can be possibly evaluated and compared with arm movements of human experimenters. Previous work in this direction is discussed in [8].

The closed-loop inverse kinematics (CLIK) scheme [17] has been used to obtain the joint variables of the robot manipulator from a differential mapping between task–space and joint–space values, denoted respectively as $\boldsymbol{p}$ and $\boldsymbol{q}$. In solving the kinematic inversion one should keep in mind that in this peculiar case, i.e. the drawing task, only the first two components of the position vector $\boldsymbol{p} = \begin{bmatrix} x & y & z \end{bmatrix}^{\mathrm{T}}$ in the task space are variable, while the z component remains constant during the task execution.
To compute the inverse kinematics we resort to the differential kinematics equation:

$$\dot{\boldsymbol{p}} = \boldsymbol{J}(\boldsymbol{q})\dot{\boldsymbol{q}} \qquad (5)$$

where $\boldsymbol{J}(\boldsymbol{q})$ is the (3×7) Jacobian matrix. This equation represent the mapping of the (7×1) velocity vector $\dot{\boldsymbol{q}}$ of the joint variables into the task space (3×1) velocity vector $\dot{\boldsymbol{p}}$. It is possible to invert the equation using the pseudo–inverse of the Jacobian matrix as follows:

$$\dot{\boldsymbol{q}} = \boldsymbol{J}^{\dagger}(\boldsymbol{q})\dot{\boldsymbol{p}} \qquad (6)$$

where $\boldsymbol{J}^{\dagger} = \boldsymbol{J}^{\mathrm{T}}(\boldsymbol{J}\boldsymbol{J}^{\mathrm{T}})^{-1}$ is a (7×3) matrix; it corresponds to the minimization of the joint velocities in a least-squares sense [17].

In order to contemplate the different characteristics of the available DOF's it could be necessary to modify the velocity distribution with respect to the least-square minimal solution. A possible solution is to consider a weighted pseudo-inverse matrix:

$$\boldsymbol{J}_W^{\dagger} = \boldsymbol{W}^{-1}\boldsymbol{J}^{\mathrm{T}}(\boldsymbol{J}\boldsymbol{W}^{-1}\boldsymbol{J}^{\mathrm{T}})^{-1} \qquad (7)$$

with $\boldsymbol{W}^{-1} = \mathrm{diag}\{\beta_1, \ldots, \beta_7\}$, where β_i is a weighting factor belonging to the interval $[0, 1]$ such that $\beta_i = 1$ corresponds to full motion for the i-th degree of mobility and $\beta_i = 0$ corresponds to freeze the corresponding joint.

Furthermore, redundancy of the robotic arm can be exploited to satisfy secondary tasks, without affecting the primary task, i.e. the motion of the drawing point $\boldsymbol{p}$. To this end, a task priority strategy [18] is used, which leads to the following solution:

$$\dot{\boldsymbol{q}} = \boldsymbol{J}_W^{\dagger}(\boldsymbol{q})\dot{\boldsymbol{p}} + \left(\boldsymbol{I}_7 - \boldsymbol{J}_W^{\dagger}(\boldsymbol{q})\boldsymbol{J}(\boldsymbol{q})\right)\dot{\boldsymbol{q}}_a \qquad (8)$$

where $\boldsymbol{I}_7$ is the (7×7) identity matrix, $\dot{\boldsymbol{q}}_a$ is an arbitrary joint velocity vector and the operator $\left(\boldsymbol{I}_7 - \boldsymbol{J}_W^{\dagger}\boldsymbol{J}\right)$ projects the joint velocity vector in the null space of the Jacobian matrix. Discrete–time integration of the joint space velocity can lead to numerical drifts; the CLIK algorithm [17] used here, allows the system to overcome this problem by exploiting the direct kinematics equation to compute an internal feedback signal from the efferent copy of the joint space variables.

The drawing task is performed on a vertical plane. Consequently, the secondary task of minimizing the gravity torques can be transformed to the joint space. This constraint provides an arm posture that is attached to the body. A possible definition of multiple secondary tasks related to the positioning of intermediate parts of the same kinematic structure, including proper trajectory planning, is presented in a more systematic fashion in [6].

IV. EXPERIMENTS AND SIMULATION

In order to test the performance of the proposed model and compare the results with human execution, we presented
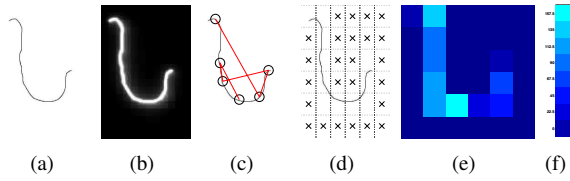
Fig. 6. The original image (6(a)), the saliency map (6(b)), and the most salient locations (6(c)) denoted by black circles. The red lines denote the scanpath that would be obtained following the approach proposed in [9]. 6(d) shows the imaginary grid superimposed on the image; cells containing an 'X' sign are those evaluated as empty. 6(e) depicts the orientation of the image patch contained in each non empty cell; the color code for orientations is explained in 6(f).
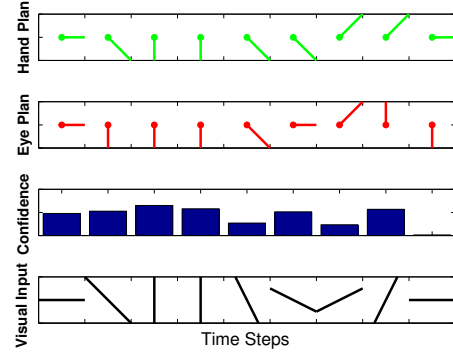

Fig. 7. The discrete–time evolution obtained as described in section II-C, with time increasing left to right. The bottom row is the sequence of visual inputs, namely the orientation of the image in the region foveated at each time step. The second bottom row shows the confidence level assigned to the eye–hand plan chosen. The two top rows depict respectively the sequences of eye movement plans, in green, and hand movement plans, in red, output by the DBN.

to a robotic simulator the same images presented to human subject in previous eye tracking sessions. In those experiments, 25 human subjects were asked to copy the images *from life*, i.e. they could look at the original image while drawing it; eye movement data were collected using an ASL 5000 eye tracker provided by the Natural Computation Lab of the University of Salerno. In Fig. 4 the plots of the fixations from four subjects can be observed; more details on the experimental setup and preliminary data analysis can be found in [19].

A. Implementation choices

The robotic arm has 7-DOF arm and the lengths of the links have been set on the basis of anatomic evaluations: 0.3 m for the first link, 0.25 m for the forearm and 0.15 m for the hand-pencil link.

In the implementation of the DBN, discrete state spaces are used for all the variables. In particular, both eye and hand movement are coded as displacement vectors, originating from the current fixation point or hand position respectively, as this is the most plausible encoding in the motor areas of primate's brain (see e.g. [20]). Recall that eye hidden variable x^e ranges in $\left\{0, \frac{\pi}{4}, \ldots, \frac{7\pi}{4}\right\}$ and the same holds for y^e, u^h, x^h and y^h. Differently, the visual feature u^e corresponds to the orientation value of the image contour in the fixated region, and ranges in $\left\{0, \frac{\pi}{8}, \ldots, \frac{7\pi}{8}\right\}$.

The problem of training the DBN consists of evaluating the probability distributions associated with hidden and output nodes, given any input configuration. In our case the variables are discrete, therefore the associated distributions are matrices, whose entries are the parameters that must be learnt. The learning technique we adopt as the basic building block is the Maximum Likelihood Estimate of the parameters using the standard Expectation Maximization algorithm for exact inference. The distributions of both the output layer (*emission* probabilities) and the hidden, unobserved layer (*transition* probabilities) are inferred from a number of *example* sequences that show how the inputs and outputs (observed nodes) are related.

The training examples we use are sequences that reflect the experimental observations on eye–tracked human subjects: hand movements are graphically continuous and correspondingly the scanpath is a coarse–grained edge–following along
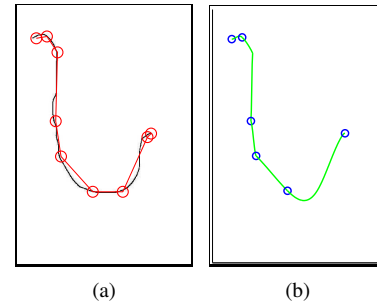

Fig. 8. The final scanpath (8(a)) and planned hand trajectory (8(b)); the blue circles in (8(b)) denote the starting and ending points of each trajectory portion. Both eye and hand movements start from the upper left corner.

the contours of the original image.

B. Results and comparisons

After running the model with the DBN trained as described above, with the input image shown in Fig 6(a), the resulting sequences of eye and hand plans $\bar{y}^e, \bar{y}^h$ are shown in the two top rows of Fig. 7. The corresponding scanpath is depicted in Fig. 8(a), and it can be directly compared to the human eye movement recordings shown in Fig. 4.
It is worth noting that a pure bottom-up, uncoupled scanpath generation would provide a very different result. This can be easily seen, for instance, by feeding the salient points in **S** to a winner–takes–all network combined with the inhibition of return as suggested in [9] in order to obtain the bottom-up fixation sequence; it is readily apparent that the sequence obtained (Fig 6(c)) is quite different from scanpaths either generated by our approach or recorded via eye-tracking; meanwhile the scanpath simulated by our system exhibits a high agreement with human performance, and work on quantitative comparisons is under way.
Fig. 8(b) shows, in green, the trajectories planned after the DBN outputs, with the endpoints evidenced by blue circles. (For human subjects, such endpoints have been found by in-
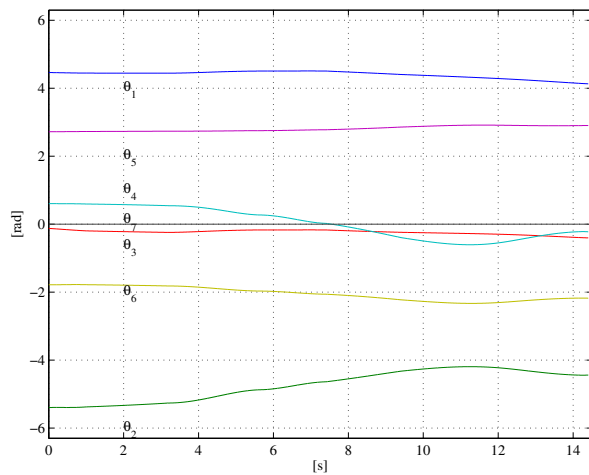
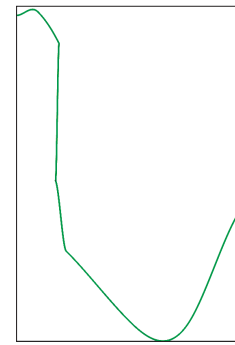Fig. 9. The time history of joint motions for the considered trajectory.

Fig. 10. The actual position of the end effector computed via direct kinematics from the joint variables. The trajectory has been translated in world coordinates considering square pixels (1 mm).

spection of the video recording, as the points where the hand interrupts for a while the drawing movement). The results of the kinematic inversion of such trajectories are shown in Fig. 9, where the time histories of the joints of the robot are depicted. Finally, Fig. 10 shows the pencil trajectory obtained by the simulated robotic arm. It can be recognized that the simulated trajectory and segmentation points are qualitatively following those recorded experimentally (Fig. 5), although direct measurements of the pencil tip, wrist and elbow would be required for a quantitative comparison of both the mobility distribution and the resulting trajectory.

V. CONCLUSIONS AND FUTURE WORKS

In this paper we presented a computational model of realistic drawing in order to investigate the issue of visuomotor coordination. The strategies adopted to coordinate such sensorimotor processes of eye and hand movement generation, during the drawing task, are inferred by a Dynamic Bayesian Network, namely an Input-Output Coupled Hidden Markov Model (IOCHMM). To the best of our knowledge such model has never been discussed before in the literature.

Experimental results are produced using 7-DOF anthropomorphic manipulator and compared to those obtained by eye-tracked human subjects involved in drawing experiments. Experiments showed that both the simulated trajectory and the gazing points have patterns quite similar to those obtained experimentally. As future work we prefigure to complete the analysis of results providing a quantitative comparison of both the mobility distribution and the final trajectory.

REFERENCES

[1] M. M. Hayhoe, D. H. Ballard, *Eye Movements in Natural Behavior*, Trends in Cognitive Science **9**, 188, 2000.
[2] B. Sheliga, L. Craighero, L. Riggio, and G. Rizzolatti, *Effects of spatial attention on directional manual and ocular responses*, Exp Brain Res, 114, 339, 1997.
[3] M. Land, N. Mennie, and J. Rusted, *Eye movements and the roles of vision in activities of daily living: making a cup of tea*, Perception, **28**, 1311–1328, 1999.
[4] K.P. Kording, D.M. Wolpert, *Bayesian decision theory in sensorimotor control.* Trends in Cognitive Sciences **10**(7), 2006.
[5] Murphy, K., Dynamic Bayesian Networks: Representation, Inference and Learning. PhD dissertation, Berkeley, University of California, Computer Science Division (2002).
[6] A. De Santis, P. Pierro, B. Siciliano *The virtual end-effectors approach for human–robot interaction*, in Roth, Lenarcic (Eds.) *Advances in Robot Kinematics*, Springer, 2006.
[7] S. Chiaverini, B. Siciliano, O. Egeland, *Redundancy resolution for the human-arm-like manipulator*, *Robotics and Autonomous Systems*, 8, pp. 239–250, 1991.
[8] V. Caggiano, A. De Santis, B. Siciliano, A. Chianese, *A biomimetic approach to mobility distribution for a human-like redundant arm*, First IEEE-RAS/EMBS International Conference on Biomedical Robotics and Biomechatronics, Pisa, I, 2006
[9] L. Itti, C. Koch: *Computational modelling of visual attention.* Nature Reviews Neuroscience **2**(3), 194–203, 2001.
[10] M. A. Goodale, G. K. Humphrey, *The objects of action and perception.* Cognition **67**, 181–207, 1998.
[11] R. Coen Cagli, P. Coraggio, P. Napoletano, *DrawBot: a bio–inspired robotic portraitist* Digital Creativity, 18, 1, 2007.
[12] S.S. Hacisalihzade, L.W. Stark, and J.S. Allen, *Visual perception and sequences of eye movement fixations: A stochastic modeling approach.* IEEE Transactions on Systems, Man, and Cybernetics, 22(3), 474-481, 1992.
[13] G.W. McConkie, P.W. Kerr, and B.P. Dyre, *What are normal eye movements during reading: Toward a mathematical description*. In J. Ygge and G. Lennerstrand (Eds.), Eye movements in reading (pp. 315327). Tarrytown, NY: Pergamon, 1994.
[14] G. Feng, *Eye movements as time-series random variables: A stochastic model of eye movement control in reading*, Cognitive Systems Research 7 (1), pp. 7095, 2006.
[15] Y. Bengio, and P. Frasconi, *Input-output HMM's for sequence processing*. IEEE Transactions on Neural Networks, 7(5), 1231-1249, 1996.
[16] S. Zhong and J. Ghosh, *HMMs and Coupled HMMs for Multi-channel EEG Classification.* In Proceedings of IEEE International Joint Conference on Neural Networks, vol. 2, pp. 1154-1159, Honolulu, Hawaii, 2002.
[17] L. Sciavicco, B. Siciliano, *Modelling and Control of Robot Manipulators*, (2nd Ed.), Springer-Verlag, London, UK, 2000.
[18] Y. Nakamura, *Advanced Robotics: Redundancy and Optimization*, Addison-Wesley, Reading, Mass., 1991.
[19] R. Coen Cagli, P. Coraggio, G. Boccignone, P. Napoletano, *The Bayesian Draughtsman: A Model For Visuomotor Coordination In Drawing.* In Proceedings of Brain Vision and Artificial Intelligence International Conference 2007, (in press).
[20] A.B. Schwartz, *Direct cortical representation of drawing movements.* Science, 265 540-543, 1994.