Article

Adaptive Behavior

Adaptive Behavior
2021, Vol. 29(6) 549–566

DOI: 10.1177/1059712320922916
journals.sagepub.com/home/adb
SAGE

Intrinsic motivation and episodic memories for robot exploration of high-dimensional sensory spaces

Guido Schillaci[1,2], Antonio Pico Villalpando[3], Verena V Hafner[3], Peter Hanappe[4], David Colliaux[4] and Timothée Wintz[4]

Abstract

This work presents an architecture that generates curiosity-driven goal-directed exploration behaviours for an image sensor of a microfarming robot. A combination of deep neural networks for offline unsupervised learning of low-dimensional features from images and of online learning of shallow neural networks representing the inverse and forward kinematics of the system have been used. The artificial curiosity system assigns interest values to a set of predefined goals and drives the exploration towards those that are expected to maximise the learning progress. We propose the integration of an episodic memory in intrinsic motivation systems to face catastrophic forgetting issues, typically experienced when performing online updates of artificial neural networks. Our results show that adopting an episodic memory system not only prevents the computational models from quickly forgetting knowledge that has been previously acquired but also provides new avenues for modulating the balance between plasticity and stability of the models.

Keywords

Adaptive models, predictive models, episodic memory, memory consolidation, intrinsic motivation, robotics

Handling Editor: Hiroyuki Iizuka, Hokkaido University, Japan

1. Introduction

Intrinsic motivation refers to the act of engaging in a pleasurable activity, where the satisfaction or reward is not coming from an external source, but from the activity itself. In psychology and robotics, this is connected to the cognitive phenomenon of curiosity, a form of intrinsic motivation that drives behaviours towards novel and surprising activity (Oudeyer et al., 2007, 2016).

Exploration and play seem to be partially driven by intrinsic motivation in infancy (Oudeyer et al., 2007). Visual exploration is likely to be one of the earliest behaviours influenced by this drive (Schlesinger, 2013), although studies in prenatal development suggest that other modalities, such as touch (Zoia et al., 2013), may be more predominant.

In developmental sciences, intrinsic motivation and curiosity are topics of great interest. It is through exploration and play–likely affected by these drives–that infants incrementally learn about their bodily capabilities and about how to interact with their surroundings (Baldassarre & Mirolli, 2013). In developmental robotics, these processes have been demonstrated to be promising tools for enabling learning, adaptivity and curiosity-driven behaviours in artificial agents (see, for instance, Colas et al., 2019; Forestier et al., 2017; Frank et al., 2014; and Cangelosi & Schlesinger, 2015, 2018; Schillaci et al., 2016, for more comprehensive reviews). Studies showed how similar mechanisms can lead an artificial system to identify states in the environment where its own actions have the greatest impact on the world as the agent perceives it (see the empowerment formalism; Salge et al., 2013).

[1]The BioRobotics Institute, Scuola Superiore Sant'Anna, Pisa, Italy
[2]Department of Excellence in Robotics & AI, Scuola Superiore Sant'Anna, Pisa, Italy
[3]Adaptive Systems Group, Humboldt-Universität zu Berlin, Berlin, Germany
[4]Sony Computer Science Laboratories, Paris, France

Corresponding author:
Guido Schillaci, The BioRobotics Institute, Scuola Superiore Sant'Anna, Viale Rinaldo Piaggio 34, 56025 Pontedera, Italy.
Email: guido.schillaci@santannapisa.it

In general, intrinsic motivation algorithms drive the actions of an artificial agent towards activities that are expected to maximise the information gain, and thus the learning progress, when this is integrated within a learning framework. These expectations are generated by predictive models, which represent the core components of such systems, allowing the agent to anticipate the sensory consequences of self-generated actions. Prediction errors can be computed as the discrepancy between the expected sensory inputs and the actual sensory observations. The system thus monitors the dynamics of such prediction errors over time and selects those behaviours that are expected to produce big variations in the prediction errors – or, in other words, those activities that may be generating a high information gain.

Traditionally, algorithms implementing intrinsic motivation have been applied in the context of learning motor control (Baldassarre & Mirolli, 2013; Baranes & Oudeyer, 2013; Oudeyer et al., 2007). This approach, combined with goal-directed exploration strategies (Rolf et al., 2012; Schmerling et al., 2015), has shown to be highly efficient when learning controllers for high-dimensional actuators. However, in such studies, the sensory space is often over-simplified or low-dimensional. For instance, it is represented as the Cartesian coordinates of the end-effector of the robot – since the main focus is rather on learning controllers of complex, high-dimensional and redundant robot manipulators.

Very few studies addressed more realistic sensory spaces (e.g. full resolution visual inputs) in the literature on intrinsic motivation algorithms for robots. Santucci et al. (2016) proposed an architecture for intrinsically motivated exploration in a humanoid robot, where the sensory space is represented by the visual input grabbed from the robot camera in a simulated environment. The authors studied how to represent goals corresponding to changes in the environment. The world consisted of a simulated environment with few rounded objects placed in front of the robot. Visual inputs and goals were encoded as binary images.

Studies that specifically address the visual modality in exploratory behaviours can be found in the wide literature on active vision. Active vision is a robotic application in which the pose and the configuration of a visual sensor is determined for solving vision-based tasks, usually those that require obtaining multiple views of an object to be manipulated (see Chen et al., 2011, for a review). Although interesting studies can be found in this literature, they mostly focus on the specific tasks to be tackled by the active visual perception and lack of generalisation to more complex and integrated sensorimotor representations. Moreover, they miss the explanatory potential that intrinsic motivation research has towards the functioning of higher cognitive processes (e.g. curiosity and learning).

There is, however, an important challenge in intrinsic motivation systems. Intrinsically motivated agents collect data and acquire skills in an incremental fashion through the online self-generation of training samples (Parisi et al., 2019). Learning is tightly coupled with the behaviour of the agent. When 'wrong' behaviours are generated in the initial phases of the learning sessions, the computational model may converge to sub-optimal solutions and local minima, thus preventing the system to generate further explorative behaviours. Adaptive systems should be escaping such situations.

Strategies to avoid this issue imply a good balance between instrumental (goal-directed) actions and epistemic (novelty-seeking) actions, so that bad-bootstrapping of models can be avoided (Tschantz et al., 2019). Different solutions have been proposed, including ϵ-greedy strategies (Oudeyer et al., 2007) (generating random actions or replaying previous experience to the system) or active inference approaches (Tschantz et al., 2019).

Related to this issue is also the problem of finding an appropriate balance between plasticity and stability of the models, when training them in an online fashion (Mermillod et al., 2013). Typically, when a model is being trained with new information, previously learned knowledge quickly becomes overwritten by the new one. This is a well-known issue in machine learning, especially in the context of artificial neural networks, named *catastrophic forgetting* (McClelland et al., 1995). In fact, the training on new samples may disrupt connection weights in the neural network that were encoding previous mappings (Masse et al., 2018). Different strategies have been proposed to overcome this problem. One of such approaches is known as *memory consolidation* or *system-level consolidation* (McClelland et al., 1995): an episodic memory system maintains a subset of previously experienced sensorimotor data and replays them, along with the new samples, to the networks during the training. Episodic memory system has been integrated recently also in the deep learning systems, such as in Deep Q-Networks implementing deep reinforcement learning (RL; Lin et al., 2018).

The work presented here approaches the aforementioned challenges by combining online[1] deep learning, intrinsic motivation and a memory consolidation system. In particular, we present an architecture that generates curiosity-driven goal-directed exploration behaviours in a robot, using computational models that can deal with high-dimensional sensory inputs. The computational models are trained in an online fashion throughout the exploratory behaviours generated by the intrinsic motivation system. Thanks to the adoption of an episodic memory, the system is less prone to catastrophic forgetting. A combination of deep and shallow neural networks has been used. In particular, deep convolutional neural networks are adopted for offline unsupervised learning of low-dimensional features from

images. Online learning is, instead, performed on shallow neural networks encoding the internal models (i.e. inverse and forward kinematics) of the robot.

A similar study can be found in the literature. In particular, Pathak et al. (2017) combine intrinsic and extrinsic rewards in a learning system that is capable of generating curiosity-driven exploration behaviours in high-dimensional sensory spaces. Authors proposed an unsupervised mechanism for learning low-dimensional features from visual inputs using a deep neural network combining convolutional and long short-term memory layers. The system is, however, applied on non-realistic visual inputs from video game environments. It is also not clear which parts of the networks are updated in an online fashion, and no report on the impact of such online updates on the networks' stability is provided (Pathak et al. 2017).

Here, we adopt an episodic memory system to face catastrophic forgetting issues experienced when performing online updates of the internal models of the robot. We present how different configurations of such an episodic memory (or the absence of it) impact overall learning progress. Moreover, our intrinsic motivation system relies on a set of goals, driving the exploration towards those that are expected to maximise the learning progress. In the experiments presented here, such goals are pre-defined and fixed, and the system switches between them according to the intrinsic motivation strategy (see section 3.2 for more details).

We test the framework on a simulator of a microfarming robot. Microfarms present interesting challenges for robotics applications. These farms are characterised by small surfaces (0.01–5 ha) and typically grow a much larger variety of crops than conventional farms. A considerable amount of work is still carried out manually, as their limited size and their diversity prevent the usage of typical agricultural machines, such as tractors. Open and lightweight robots for microfarms may reduce manual labour and increase their productivity. However, the robots must be able to cope with many types of plants and dense plant populations.

A first step in the development of such advanced applications is to automatically construct a three-dimensional (3D) representation of plants in outdoor settings. The 3D scanning procedure collects many images of the object of interest using either multiple fixed cameras or a single one that moves around the object along a pre-defined trajectory. The image set is converted into a point cloud (e.g. as in structure-from-motion algorithms; Schoenberger et al., 2016; Schoenberger & Frahm, 2016) and then post-processed for estimating 3D models or for detecting plant organs.

Plants are, however, complex objects to reconstruct and many parts are hidden, for example, by leaves. If the camera movements are not sufficiently accurate to uncover such hidden spots, the reconstruction process may generate incomplete 3D models. The trajectory of the camera movements must therefore adapt to the plant of interest. Intelligent and adaptive behaviours – like those generated by the proposed framework – could maximise the information captured by the robot camera.

The rest of this article is structured as follows. First, we introduce the robotic platform (section 2.1) and the simulator (section 2.2) that have been used in this study. In section 3, we describe the learning architecture, including the computational models, the goal selection strategy, the image encoding and the main learning algorithm. Sections 4 and 5 describe the experiments and the results, respectively. In particular, we show the performance of the system under different configurations, giving a particular emphasis on the episodic memory system. We draw our conclusions in section 6, where we present the plan for future work.

2. Experimental setup

2.1. Robotic platform

This work presents an architecture that generates curiosity-driven goal-directed exploration behaviours for an image sensor of a microfarming robot. The studies reported here have been carried out on a simulated version (described in section 2.2) of the LettuceThink microfarming robot (Figure 1) developed by Sony Computer Science Laboratories. The platform consists of an aluminium frame with an X-Carve computer numerical control (CNC) machine mounted on it. The CNC machine is used to provide 3-axes movements to a depth camera (Sony DepthSense) mounted at the tip of the vertical z-axis (hereafter, the *end-effector camera*). In the experiments presented in this article, the end-effector camera is facing top-down and only two motors are used (x and y). This configuration is not intended to be used for 3D reconstruction of plants,

Figure 1. The LettuceThink microfarming robot.

which requires more degrees of freedom for the camera movements.

A Raspberry Pi embedded computer connects to the X-Carve controller over a serial connection and to the end-effector depth camera over USB. The X-Carve controller consists of an Arduino board running the Grbl firmware, an open-source software that interprets G-code instructions and controls the stepper motors of the CNC machine.[2] The stepper motors are not equipped with rotary encoders but their status are obtained through a Grbl WPos (working position) request that returns the offset of each motor from the initial position by counting the steps performed from the beginning of the operation. The Raspberry Pi uses the ROS (robot operating system) middleware to let external computers control the robot hardware and to synchronise the motor and image data.

2.2. Robot simulator

A simulator of the LettuceThink robot has been developed to ease the testing of different configurations of the learning system. The simulator generates sensorimotor data from requested trajectories of the end-effector camera. Knowing the initial position of the CNC machine and the target position, the simulator linearly interpolates the trajectory and returns the intermediate positions of the camera together with the images captured from each specific position. The sensorimotor data returned by the simulator have been pre-recorded by performing a full scan of the (x, y) plane of the CNC machine using a resolution of 5 mm. This resulted in 24,964 images, each mapped to an (x, y) position of the CNC machine.

The simulator is freely available as a python script that generates sensorimotor data[3] from the recorded dataset.[4] All the software developed for this work is freely available in an online repository.[5]

3. Learning architecture

The architecture implementing the curiosity-driven goal-directed exploration behaviours combines one deep and two shallow neural networks, as well as an episodic memory system. A deep convolutional neural network is trained offline using unsupervised methods to encode images using low-dimensional features. The shallow neural networks learn online how to represent the inverse and forward kinematics of the robotics system. In addition, an artificial curiosity system assigns interest values to a set of pre-defined goals and drives the exploration towards goals that are expected to maximise the learning progress. An *episodic memory system* is included to handle catastrophic forgetting issues.

The learning architecture controls the movements of the end-effector camera of the LettuceThink platform using an exploration behaviour driven by artificial curiosity. Differently to adopting pre-defined trajectories, the behaviours generated by this system aim at autonomously maximising the information obtained with the image sensors. In particular, the aim of the learning architecture is to direct the movements of the camera towards positions that produce informative views of objects of interest, and in parallel to update the internal models with the sensorimotor data generated by the latest movements.

The architecture is illustrated in Figure 2 and is partially inspired by the intrinsic motivation algorithm presented by Oudeyer et al. (2007) and on our previous works on goal-directed exploration (Schmerling et al., 2015). This article proposes more advanced computational models compared to those presented by Oudeyer et al. (2007), as well as more adaptive exploration strategies than those adopted in Schmerling et al. (2015).

3.1. Forward and inverse models

The architecture is based on two internal models (i.e. forward and inverse models) which encode the dynamics of the sensorimotor system of the robot. Such models are not pre-programmed, but rather trained in an online fashion.

The forward and inverse models are implemented as shallow artificial neural networks that link motor commands to visual inputs and vice versa. The sensory space here is identified as the space of images that can be grabbed from the end-effector camera of the robot. The motor space is identified as the space of the CNC (x, y) motor positions. We define a goal as a specific state in the sensory space. When the system is given a goal (i.e. a target image), the inverse model is queried to generate the best possible motor commands to reach this goal–that is, to move the camera to the position where the goal image has been recorded. A copy of this motor command is sent to the forward model. The forward model internally simulates the visual input as if the movement were executed.[6]

Both models are learned and updated in runtime in an incremental fashion. At the beginning of the learning session, the inverse model – whose weights are randomly initialised – is likely to generate random motor commands. This movement produces sensorimotor data that are used to update the model on the fly. Learning is thus coupled with behaviour, meaning that the training data are produced by the behaviours generated by the models that are being updated – in parallel – during the activity.

The system calculates a prediction error PE by comparing the image predicted by the forward model and the sensory observation captured from the visual input after the execution of the movement. The dynamics of the prediction errors recorded during each attempt to reach a specific goal are monitored. In particular, we use the trend (i.e. the derivative) of the prediction error

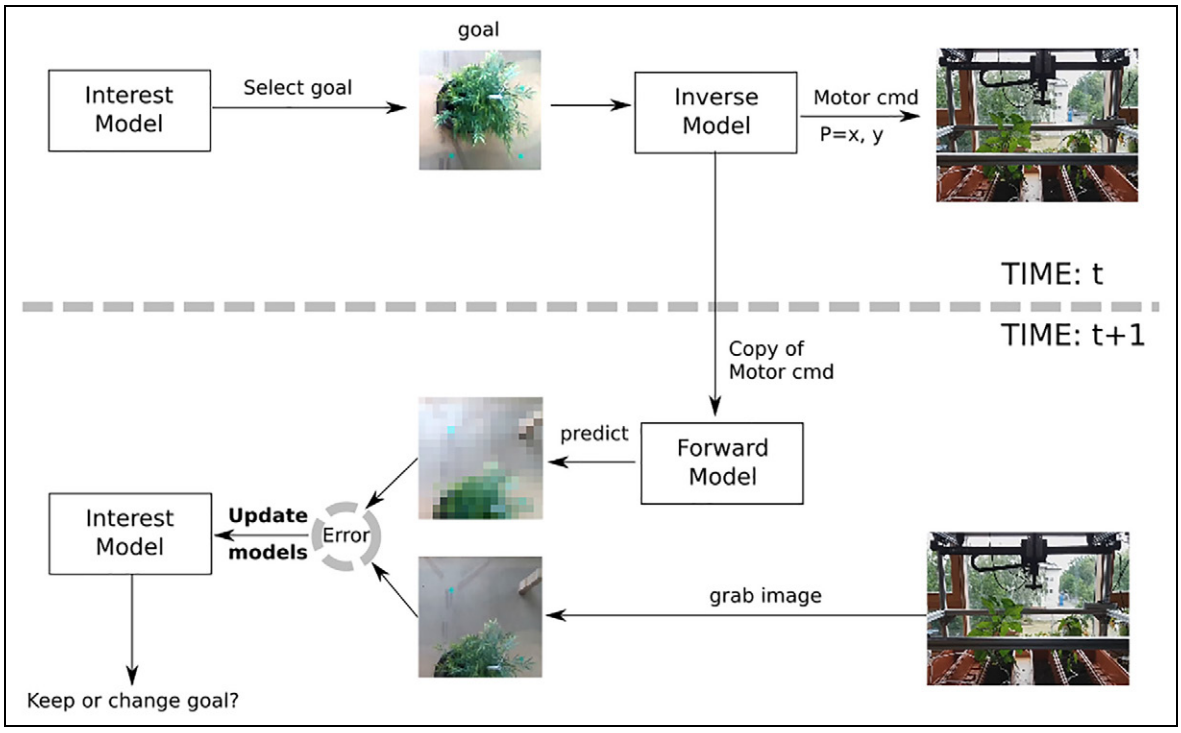


Figure 2. The architecture that generates goal-directed movements driven by artificial curiosity. In the illustration, the system selects a goal (an image) and encodes it into a lower-dimensional features vector using the convolutional autoencoder. The compressed goal is fed into the inverse model, which infers the required motor command to reach it. An efferent copy of the motor command is fed into the forward model, which estimates the sensory input that would be observed after the execution of the command. Once the action is performed and the new sensory observation is available, this is compared to the prediction, resulting in a prediction error. Prediction error is thus used to update the learning progress of the current goal. The system thus decides whether to keep exploring this goal or to switch to another, according to their expected learning progress. The sensorimotor data gathered throughout the exploration behaviour are used to update the inverse and forward models, and the episodic memory, in an online fashion. See section 3.5 for further details.

as an indicator of the expected learning progress: the larger the *change* in the prediction error, the bigger the expected learning progress. In other words, big changes in the mismatch between what the system expects to perceive and what it observes are interpreted as cues for novelty.

3.2. Goal selection and learning progress

At the beginning of the learning process, the system is given a pre-defined set of goals (i.e. a set of images recorded from different camera positions). At every iteration, an *interest model* chooses a goal according to a goal selection strategy. This strategy relies on the estimated learning progress that is measured for each goal. That is, the system maintains for each goal an indicator about whether it is advantageous – in terms of information gain – to continue exploring around this goal or not. If exploring a goal is not expected to produce big changes in the prediction error (i.e. the expected learning progress is low), then the interest model drives the exploration towards another interesting goal.

The expected learning progress LP of a goal g, that is, LP_g, is computed as follows

$$\mathrm{LP}_g = \tanh(|\mathrm{PE}_g(t) - \mathrm{PE}_g(t-1)|) \quad (1)$$

where $\mathrm{PE}_g(t)$ is the prediction error calculated at time t. At every iteration, the system chooses with a probability of 15% a greedy goal selection strategy, instead of the aforementioned one (see Oudeyer et al., 2007, for the rationale behind this approach, and Tschantz et al., 2019, for alternative approaches). This strategy selects a random goal from the existing set of goals.

Moreover, with a probability of 30%, a random movement is generated instead of the predicted camera movement. This is performed to prevent the system from converging towards local minima.[7]

3.3. Image encoding

Using images as sensory states raises the issue about how to measure dissimilarity between predictions and observations. Images are high-dimensional data. Typically, in machine learning, dissimilarity between

images is not computed by pixelwise comparisons, but rather by comparing features extracted from them (Chen & Chu, 2005). In this work, we used an unsupervised learning technique to extract low-dimensional features from images, so that the resulting features can be simply compared using Euclidean distances. We adopted deep convolutional autoencoders (CAEs) for this task. A CAE (Masci et al., 2011) is a deep neural network that is trained to reproduce the same data passed as input. The dimensionality of the layers in the network decreases from the input layer to a central one (the *latent layer*) and then increases back to the original dimensionality at the output layer. This enables the learning of low-dimensional features in the latent layer, representing the input data. Autoencoders are typically used for compressing high-dimensional data, such as raw images, into low-dimensional codes, and eventually for reconstructing the input from these features.

Autoencoders generally have fully connected layers. CAEs extend the original structure of autoencoders using convolutional layers. In convolutional layers, the connectivity between neurons resembles the organisation of the human visual cortex, where neurons respond only to limited parts of the visual field, known as the *receptive field*. Besides being more biologically plausible, convolutional layers considerably reduce the number of parameters that need to be learned by the artificial neural network and have been demonstrated to perform much better in computer vision applications.

The usage of the CAE allows a simpler calculation of prediction errors. Calculating prediction errors corresponds to computing the Euclidean distance between the compressed representations of the predicted image and the observed one.

3.4. Episodic memory

As mentioned in the introduction of this article, training artificial neural networks in an online fashion may produce catastrophic forgetting issues. In the current architecture, the shallow neural networks used in the forward and inverse model are trained on new tasks on the fly. Such training on new samples may disrupt connection weights that were encoding previous mappings (Masse et al., 2018). Different strategies have been proposed to overcome this problem, and the literature on catastrophic forgetting is rich of approaches. Parisi et al. (2018) propose an interesting architecture with a dual memory system: an episodic memory that learns representations of sensory experience in an unsupervised fashion through input-driven plasticity, and a semantic memory develops more compact representations of statistical regularities embedded in episodic experience. Both memories are implemented as growing recurrent networks. A dual memory system has been proposed recently also in the context of deep generative models by Kamra et al. (2018). Interestingly, this approach adopts generative replays of past experience.

To the best of our knowledge, episodic memory systems have not been integrated within intrinsic motivation systems and for exploration behaviours on high-dimensional sensory spaces. Perhaps the closest studies can be found in the RL literature, for instance by Isele and Cosgun (2018), which proposes different strategies for selecting which experiences will be stored in an episodic memory buffer. We adopt a similar episodic memory system, which maintains a subset of previously experienced sensorimotor data and replays them, along with the new samples, to the networks during the training. This approach is known as memory consolidation or system-level consolidation (McClelland et al., 1995).

The episodic memory is fed with samples observed during the behaviour of the artificial agent. In particular, the episodic memory is empty at the beginning of the learning session. New samples, as observed from the camera and from the motor system (both sensorimotor information are stored into a memory element), are added into the memory, as soon as this reaches its full size (as described in section 4, we vary this parameter in our experiments). When the memory is full, the system keeps adding new samples into it, whenever available, discarding old elements of the memory. A greedy strategy is adopted, where random elements of the memory are removed from the list and replaced with new ones. Moreover, a second parameter drives such an episodic memory update: p_{em}, that is, the probability of changing an element in the episodic memory. The episodic memory update process consists of iterating over all the elements of the memory and, with a probability of p_{em}, of discarding each element and replacing it with the new sample. This produces duplicates of the new samples in the memory, which may increase the plasticity of the system towards latest observations.

In sections 4 and 5, we show the impact of different configurations of the episodic memory to the whole learning process and to the behaviour of the simulated robot.

3.5. The learning algorithm

Using the components described above, we can now describe the basic learning algorithm as follows:

1. The system selects N goals, g_i, following the strategy described in section 3.2. Each goal is the compressed representation of an image produced by the pre-trained CAE. The prediction errors $\mathrm{PE}(\mathrm{g}_i, 0)$ are initialised to 0, and $\mathrm{LP}(\mathrm{g}_i, 0)$ to 0, for each goal.
2. The interest model selects the goal $\mathrm{g}(t)$ in $\{\mathrm{g}_i\}$ which has the maximum expected learning progress $\mathrm{LP}(\mathrm{g}_i, t)$. The first goal $\mathrm{g}(0)$ of the learning process is randomly selected from $\{\mathrm{g}_i\}$.

3. The inverse model computes the new position of the camera P(t) with the objective that the (encoded) image of the camera after the movement corresponds to the selected goal.
4. The position P(t) is sent to the forward model. The forward model predicts the compressed image $I_p(t)$ that will be seen at the new position.
5. The camera is moved. A random noise is added to the camera movement predicted by the inverse model, in order to produce small fluctuations and thus to generate more views of the target locations.
6. The camera image $I_c(t)$ is grabbed and compressed by the CAE.
7. The prediction error PE(t) is computed as the Euclidean distance between $I_p(t)$ and $I_c(t)$ (the compressed representations of the predicted and current images).
8. The estimated learning progress is updated using equation (1) and PE(t).
9. The inverse model is updated with the recorded sensorimotor data and the elements stored in the memory.
10. The forward model is updated with the recorded sensorimotor data and the elements stored in the memory.
11. Update the episodic memory, as described in section 3.4.
12. Return to step 1, until the maximum number of iterations is reached (5000 in the current experiment).

When the camera moves to a new position computed by the inverse model, the difference between the expected image and the obtained image may be very large. The prediction error PE will be large in that case. If the previous action of reaching the same goal produced a prediction error of different magnitude, this results in a high estimated learning progress LP: the system's guess about the consequences of the current action is not met, and it also changes very much when re-iterating it, suggesting that such an action is very informative. The interest model is then likely to select the same goal for the next iteration (step 2). In case the current activity does not produce big changes in the prediction error, compared to the previous ones, the interest model selects another goal, as the current one does not bring any novel information to the learning process.

It has to be noted that we do not update the inverse and forward models at each iteration but only after a given batch size has been completed. The fitting is then done using the history of sensory and motor data for that batch. We use batches for performance reasons and for preventing less accurate gradient estimations due to the big fluctuations that would be produced, otherwise, by stochastic learning (i.e. batch size of one sample).

4. Experiments

As mentioned in the previous sections, we adopt a CAE for unsupervised learning of image features. The CAE is pre-trained in an offline fashion that is ahead of the learning experiment. Thereafter, during the online test, image goals are compressed into features, using the encoder part of the pre-trained CAE.

The CAE has been implemented using Keras deep learning library and TensorFlow backend. The structure of the CAE is the following. An input layer takes a 64×64 grayscale image and passes it to a sequence of conv2D and MaxPooling2D layers: conv2d_1 has $64 \times 64 \times 256$ neurons, conv2d_2 has $32 \times 32 \times 128$ neurons and conv2d_3 has $16 \times 16 \times 128$ neurons. The output of conv2d_3 is flattened and connected to a Dense layer of size 32 (the size of the latent space, that is, the feature vector representing the compressed image). The decoder part of the CAE starts with connecting the latent later to a Dense layer of size 64, reshaping it into a 16×16 layer and connecting it to a conv2d layer, characterised by $8 \times 8 \times 32$ neurons. The following sequence of layers is then expanding the latent signal: upsampling2d, conv2d($16 \times 16 \times 128$), upsampling2d, conv2d($32 \times 32 \times 512$), upsampling2d, conv2d($64 \times 64 \times 1$). The latter represents the output *decoded* layer, whose size matches the one of the input. The MaxPooling kernel has size 2, padding 'same'. All the conv2D layers have kernel size equals to 3 and ReLU activation functions except the CAE output layer that has a sigmoid activation function. An ADAM optimiser has been used for training the network on an MSE loss function.

The inverse and forward models have been implemented as shallow artificial neural networks and are trained in an online fashion during the experiment. The structure of the forward model is characterised by the following layer. An input layer is fed with a two-dimensional vector (the (x, y) motor command applied to the CNC machine), which is expanded to a dense layer of 32 neurons, followed by two dense layers of 320 neurons, and a dense layer of 32 neurons (the latent code dimensionality), representing the output of the forward model. The structure of the inverse model is characterised by the following layers. An input layer is fed with a 32-dimensional vector (the latent code, representing the goal image), which is passed to a dense layer of the same size, then expanded to two dense layers of 320 neurons, and compressed back to a dense layer of 2 neurons, representing the output of the inverse model (the (x, y) motor command to apply to the CNC machine). For both the inverse and forward models, the dense layers have tanh activation functions. A standard gradient descent (LR: 0.0014, decay: 0.0, momentum: 0.8) is used as optimiser on an MSE loss function. The training of the CAE, the inverse and forward model updates, as well as the predictive processes

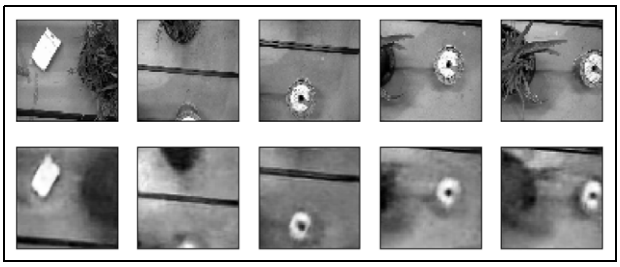


Figure 3. Images encoded and decoded using the convolutional autoencoder. The top row shows the original images. The bottom row shows the result of encoding each input image into a latent code and then decoding it back to the original image resolution.

are run on a machine equipped with an NVIDIA graphics card.

As discussed in the previous sections, we adopt an episodic memory system that maintains a subset of previously experienced sensorimotor data and replays them, along with the new samples, to the networks during the training. As soon as they are available, new samples (image and motor information) are added into the memory. When the memory reaches its full size, random elements are removed from the memory and replaced with new ones (see section 3.4 for more details).

The inverse and forward models are updated at constant frequency, that is, every time a new buffer of 16 sensorimotor samples (each consisting of (x, y) motor commands and 32-dimensional compressed image codes gathered along the exploration) is available. In particular, a new fit for both the inverse and forward models is triggered, passing as training data the buffer with the *new* 16 observations together with *all* the samples stored in the episodic memory.

In the current experiment, we vary two parameters over different runs and analyse their influence onto the learning process. In particular, we vary the size of the episodic memory and the probability of changing an element in the episodic memory (p_{em}).

5. Results

As discussed in the previous section, the experiment consisted of two phases. In an initial offline stage, a CAE has been trained with the images stored in the simulated robot data (24,964 images, see section 2.2). This neural network has been used to compress the images (sensory inputs and goals) to be used in the online learning process.

Figure 3 shows an example of a sequence of encoded-decoded images produced by such a CAE. As mentioned in the previous section, the image features (i.e. the compressed version of the image) consists of a 32-dimensional vector. Input images and reconstructed images are characterised by a resolution of 64×64 pixels. The reconstructed images shown in Figure 3 have been generated by a CAE trained for 50 epochs.

Once having trained the CAE, the full learning process can be performed as described in section 3.5. We carried out a series of experiments, in which we varied two parameters of the system: the size of the episodic memory mem_{size} and the probability of updating the memory elements p_{em}. In particular, we tested three different values for $\text{mem}_{\text{size}} : \{0, 10, 20\}$ (with 0 meaning that no episodic memory available) and two values for $p_{\text{em}} : \{0.1, 0.01\}$, for a total of six experiments. The values of mem_{size} specify the number of batches (each batch has 16 sensorimotor samples) that are contained in the memory. Each experiment has been run five times, for a total of 30 runs. Each run consisted of 5000 iterations of the algorithm described in section 3.5. During each run, the mean squared error (MSE) of the forward and inverse models, calculated on the same test dataset (consisting of 50 images randomly chosen from the simulator dataset), have been monitored. The MSEs have been calculated every 50 iterations of the algorithm (behaviour and model updates). A set of nine goal images have been chosen from the simulator dataset.

Figure 4 shows an example of a run of the exploration behaviour under a specific configuration ($\text{mem}_{\text{size}} = 20$, $p_{\text{em}} = 0.1$) and over different time steps (50, 1000, 2000, 3000, 4000 and 5000 iterations). The axes specify the x and y motor positions of the CNC machine. Red dots represent the motor projections of the nine image goals (ground truth data from the simulator dataset). Green dots represent the explored positions of the robot. As it can be seen, the behaviour of the robot tends towards exploring around the regions of the goals. The effect of the ϵ-greedy strategy is also evident from the almost uniformly distributed green points within the action space. In other works, the exploration tends to visit also regions far from the goals as a result of the greedy exploration strategy (random commands) which is applied with a certain frequency. The goal positions in the 32-D space are compressed representations of the original goal images. What is visualised as 'goals' (red dots) in Figure 4 are the ground truth motor position mapped to the original goal images. We believe that achieving the targets requires so many iterations because the learning system has to cope with the errors introduced with the CAE compression and with learning the mapping between the latent goal codes and the right motor positions. Moreover, the mapping is being learned over time with the sensorimotor data generated with the updating models. During the initial bootstrapping, the models may not have observed any sensorimotor mapping useful for reaching specific goals. Therefore, the networks are not capable of generating correct motor commands to reach them, nor can they learn how to do that if no

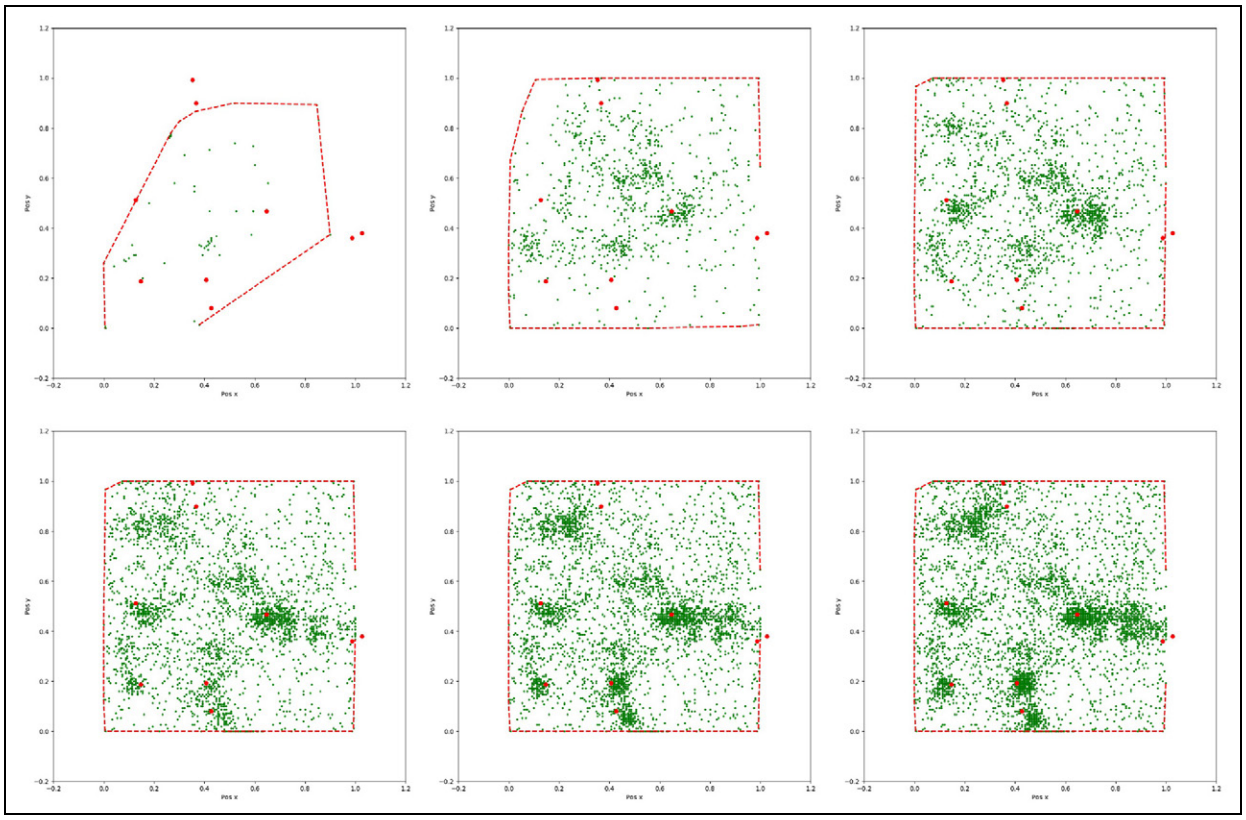


Figure 4. An example of a goal-directed exploration session driven by artificial curiosity (run no. 5, configuration ($mem_{size} = 20$, $p_{em} = 0.1$)). A set of nine goals are defined from the pre-recorded image dataset. The red dots show the (x, y) motor configurations – which are stored as ground truth data in the dataset – that correspond to the image goals. Each of the plots shows the experienced (x, y) motor commands after 50 (top left plot), 1000, 2000 (top right), 3000 (bottom left), 4000 and 5000 (bottom right) samples. Green dots represent the (x, y) motor positions captured from the CNC machine while exploring. The red dashed line shows the convex hull computed around the explored points. The exploration tends to visit also regions far from the goals as a result of the greedy exploration strategy (random commands) which is applied with a certain frequency.

relevant activity towards those goals has been observed. Eventually, noise and greedy exploration may introduce the required experience to achieve them, but this takes time. Further work will analyse this aspect of the learning process.

Figure 5 shows the predictions of the inverse models over time. The inverse model is fed with the encoded goal image as input and returns as output the (x, y) coordinates of the motor. It can be clearly seen that the learning process brings the prediction towards the target motor goals (ground truth data). The episodic memory makes sure that previous knowledge (when alternating between goals) is not forgotten. The impact of catastrophic forgetting, due to the absence of the episodic memory, is more clear in Figure 6, which shows the prediction of the inverse model over time. As it can be seen, predictions over different goals are much more noisy than in the previous case.

Figure 7 shows the dynamics of the expected learning progress over the nine goals of the same sample run described above ($mem_{size} = 20$, $p_{em} = 0.1$). Each goal is characterised by the expected learning progress described in equation (1). A decay factor is also added, so that expected learning progress slowly decays over time. The functional role of the decay factor for the expected learning progress is to prevent that goals not being explored maintain the same interest value. Moreover, the expected progress becomes zero for a goal that has not been adopted for a longer time period. Figure 8 shows a zoomed-in version of the first goal plot of Figure 7, together with a zoomed-in version of Figure 5. Goal 0 is selected during these iterations. As it can be seen, the corresponding learning progress is changing over time – that is, prediction error calculated on this goal is varying. The effect can be seen on the predictions of the motor position, visualised in the bottom plot of 5, which improve over time.

The time scales (horizontal axis) of this plot and those of Figures 5 and 6 are not matching as predictions are performed only when the specific goal is selected by the intrinsic motivation system.

Figure 9 compares the learning progress of the forward model in three different configurations of the episodic memory. As it can be seen, the intrinsic

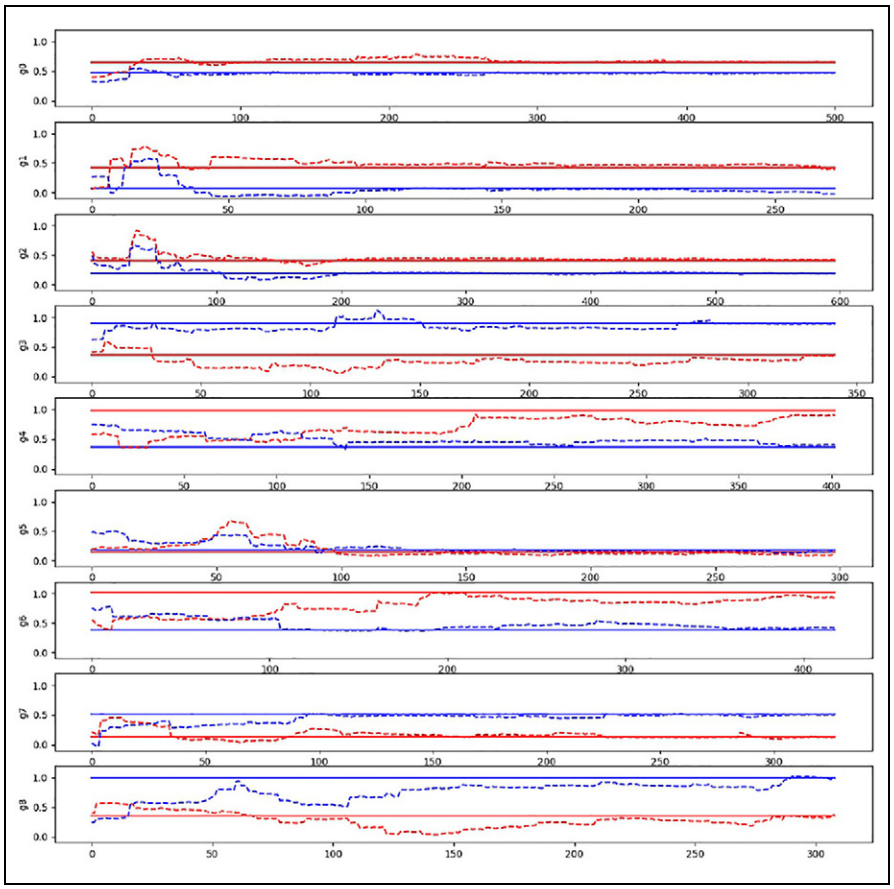


Figure 5. The predictions of the nine goals over time performed by the inverse model in the run no. 5 of the configuration ($mem_{size} = 20$, $p_{em} = 0.1$). The inverse model is fed with the encoded goal image as input and returns as output the (x: blue, y: red) coordinates of the motor. Solid lines show, for each goal image, the ground truth motor positions. Dashed lines show the predictions of the inverse model.

motivation system does not work well without episodic memory (red curve). Adding an episodic memory has a positive impact on the learning progress, as the MSE of the forward model predictions decreases over time (bigger memory generates better results). Similar trends can be observed for the inverse model (Figure 10).

Reducing the probability of changing memory elements to $p_{em} = 0.01$ produces smoother descending trends in the MSE of the configurations with episodic memory, for both the forward (Figure 11) and the inverse (Figure 12) models. We explain this effect as a result of a smaller level of plasticity in the episodic memory. As explained in the previous section, the current episodic memory update strategy produces duplicates of the new observations in the memory. Higher values of p_{em} produces more duplicates of the same samples and may reduce the variance within the episodic memory.

This can also be seen when comparing the MSEs of the forward and inverse models by pivoting on the p_{em} values, as shown in the following figures. In Figure 13, for instance, it can be seen that the system with higher probability of memory updates (red curve) has a steeper descending curve, although on the long run is outperformed by the other strategy (both the configuration are characterised by a memory size of 10 batches). This suggests that updating the episodic memory more quickly produces higher plasticity, whereas doing it more slowly produces more stability. This effect is, however, not clearly visible in the MSE curves of the inverse model (Figure 14). Increasing the memory size to 20 batches leverages this different effect in the forward model, as it can be seen from Figure 15, but makes it slightly more evident in the inverse model MSE trends (Figure 16).

Further studies will focus on a deeper analysis of the variance of the elements in the episodic memory, as well as on more sophisticated strategies for memory update and their effect on the plasticity and stability of the models.

Dynamic environmental conditions have not been addressed in this study. In a recent work (Miranda & Schillaci, 2019), we show that a similar episodic memory system can be used in transfer learning, as well.

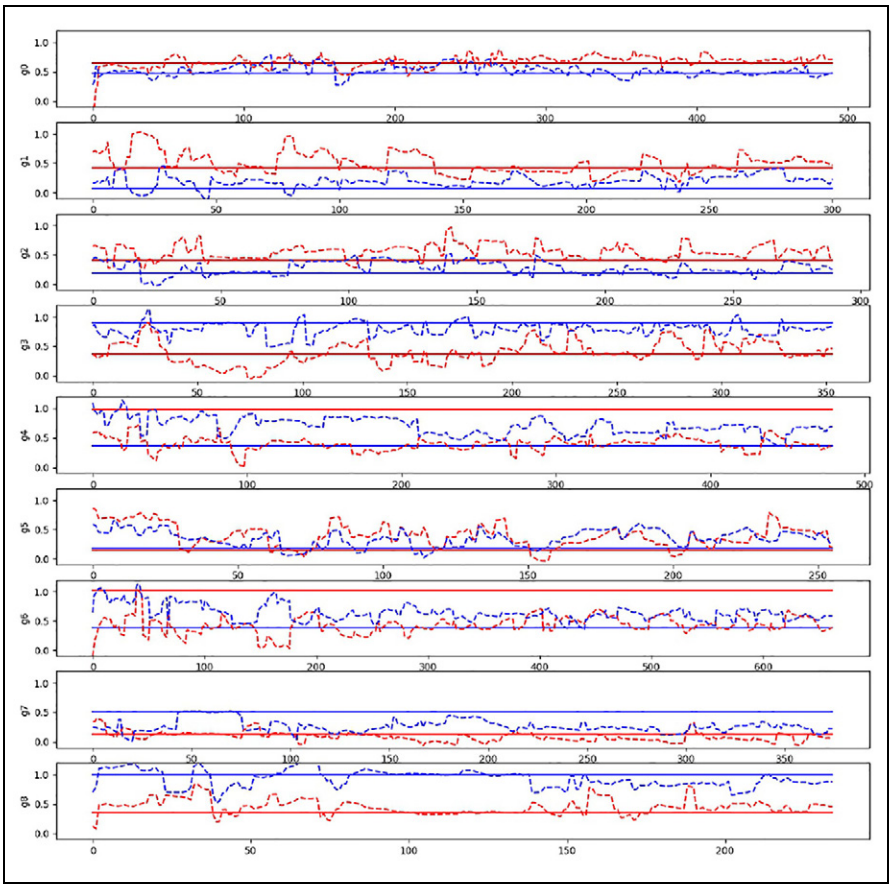


Figure 6. The predictions of the nine goals over time performed by the inverse model in the run no. 0 of the configuration without episodic memory. The inverse model is fed with the encoded goal image as input and returns as output the (x: blue, y: red) coordinates of the motor. Solid lines show, for each goal image, the ground truth motor positions. Dashed lines show the predictions of the inverse model.

Although applied on a different experimental setup and computational models, we demonstrate that the episodic memory enables adapting to changing environmental conditions maintaining good model performance on previous conditions. Future work will analyse the performance of the system presented here in the context of dynamic environments.

6. Conclusion

We presented a learning architecture that generates curiosity-driven goal-directed behaviours based on intrinsic motivation and on an episodic memory system. In particular, we implemented online learning mechanisms on artificial neural networks using an episodic memory system for facing catastrophic forgetting issues. We adopted and trained a CAE for compressing image goals and observations into low-dimensional features, which can be more easily processed by our models. We showed the performance of our system under different configurations, where plasticity and stability of the learning process can be modulated. Of fundamental importance were the predictive processes implemented through the forward models. In fact, processes of anticipation of sensorimotor activity have enabled the advanced behaviours showed in our experiments. We strongly believe that similar mechanisms may represent a bridge between motor and cognitive development in humans and a promising tool for cognitive robotics. We tested the models on simulated sensorimotor data from a microfarming robot. This article has contributed also to the widening of the developmental robotics approach towards applications and robotic platforms that are non-conventional in this field.

Our experiments have shown that the intrinsic motivation systems can work also in the presence of a high-dimensional sensory space. Adopting an episodic memory system not only prevents the computational models to quickly forget knowledge that has been previously acquired, but also provides new avenues for modulating the balance between plasticity and stability of the system. In deep RL, the adoption of episodic memories has also shown to mitigate slowing factors –

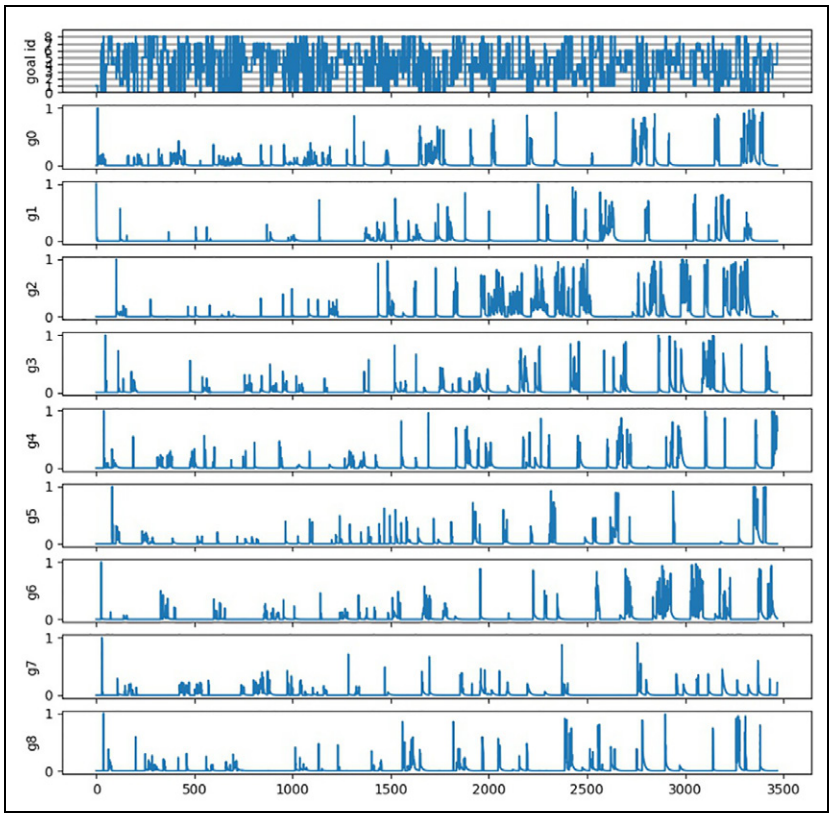


Figure 7. The dynamics of the expected learning progress for each goal (from 2nd to 10th plot). The first plot shows the ID of the currently selected goal over time. Rows g0 to g8 show the dynamics of the learning progress for each of the nine goals, calculated using equation (1). Note, from the equation, that the hyperbolic tangent calculated on a positive value (absolute value of the difference between two subsequent prediction errors) has a range between 0 and 1.

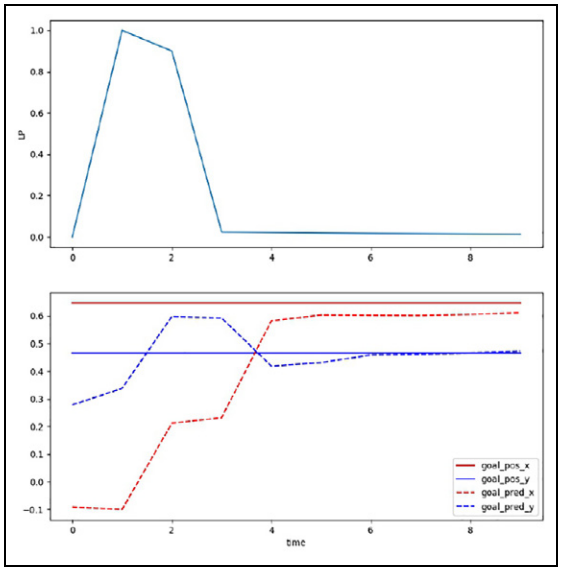


Figure 8. These plots illustrate zoomed-in parts of the plots in Figures 7 and 5. In particular, the plot on the top shows the learning progress for the goal 0 (second plot in Figure 7) during the first 10 iterations of the exploration. The bottom plot shows the predictions of the motor positions corresponding to goal 0 (first plot in Figure 5) during the first 10 iterations of the exploration.

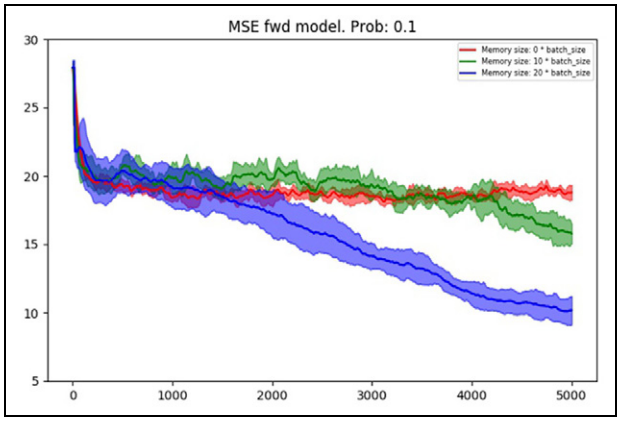


Figure 9. The mean squared error of the forward model computed over 5000 iterations, averaged on 5 runs. The plot compares the MSE of three configurations of the system: without memory (0 batches), $\text{mem}_{\text{size}} = 10$ and $\text{mem}_{\text{size}} = 20$. In this plot, the probability of changing memory elements is set to $p_{\text{em}} = 0.1$. Solid lines show the mean of the MSE over 5 runs, for each configuration. Shaded areas show the (mean $-$ stddev, mean $+$ stddev) areas.

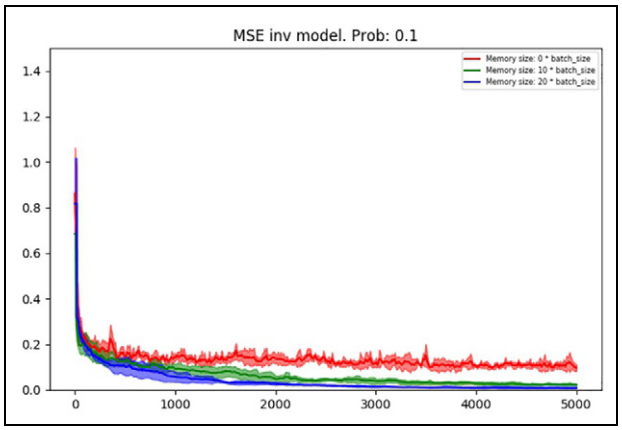


Figure 10. The mean squared error of the inverse model computed over 5000 iterations, averaged on 5 runs. The plot compares the MSE of three configurations of the system: without memory (0 batches), $\text{mem}_{\text{size}} = 10$ and $\text{mem}_{\text{size}} = 20$. In this plot, the probability of changing memory elements is set to $p_{\text{em}} = 0.1$. Solid lines show the mean of the MSE over 5 runs, for each configuration. Shaded areas show the (mean − stddev, mean + stddev) areas.

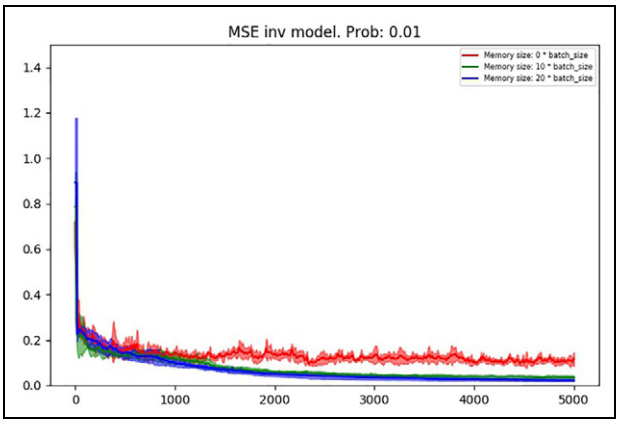


Figure 12. The mean squared error of the inverse model computed over 5000 iterations, averaged on 5 runs. The plot compares the MSE of three configurations of the system: without memory (0 batches), $\text{mem}_{\text{size}} = 10$ and $\text{mem}_{\text{size}} = 20$. In this plot, the probability of changing memory elements is set to $p_{\text{em}} = 0.01$. Solid lines show the mean of the MSE over 5 runs, for each configuration. Shaded areas show the (mean − stddev, mean + stddev) areas.

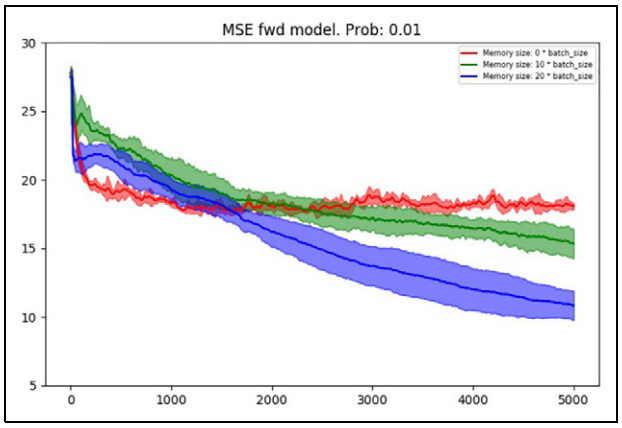


Figure 11. The mean squared error of the forward model computed over 5000 iterations, averaged on 5 runs. The plot compares the MSE of three configurations of the system: without memory (0 batches), $\text{mem}_{\text{size}} = 10$ and $\text{mem}_{\text{size}} = 20$. In this plot, the probability of changing memory elements is set to $p_{\text{em}} = 0.01$. Solid lines show the mean of the MSE over 5 runs, for each configuration. Shaded areas show the (mean − stddev, mean + stddev) areas.

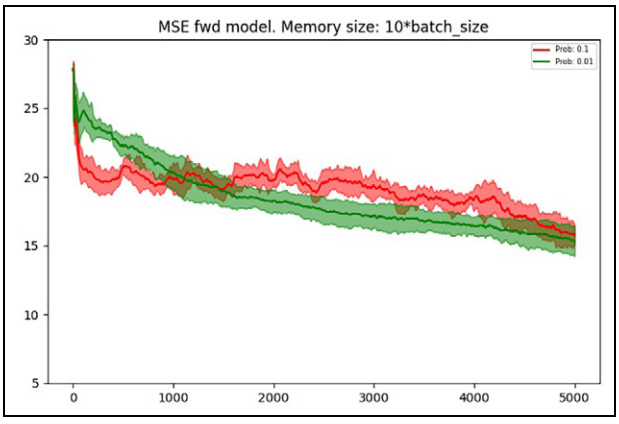


Figure 13. The mean squared error of the forward model computed over 5000 iterations, averaged on 5 runs. The plot compares the MSE of two configurations of the system: $p_{\text{em}} = 0.1$ and $p_{\text{em}} = 0.01$. In this plot, the episodic memory size is set to 10 batches. Solid lines show the mean of the MSE over 5 runs, for each configuration. Shaded areas show the (mean − stddev, mean + stddev) areas.

namely incremental parameter adjustment and weak inductive bias – and the original demands for huge amounts of training data, effectively allowing deep RL to be faster (Botvinick et al., 2019). We expect episodic memories to provide a similar contribution to intrinsic motivation systems, being IM conceptually related to RL. Nonetheless, future investigations should be carried out in support to this claim.

Moreover, the aim of this work was not to provide further evidence that intrinsically motivated goal-directed exploration behaviours outperform random exploration behaviours in the bootstrapping of motor control. Therefore, we have not presented any comparison between the performances of the proposed mechanism and those of greedy exploration strategies. Several related studies can be found in the literature, as discussed in the introduction of this article. These studies typically address robotic platforms whose actuators are characterised by higher number of degrees of freedom, compared to the platform adopted in this study – the Sony LettuceThink robot. The positive contribution of

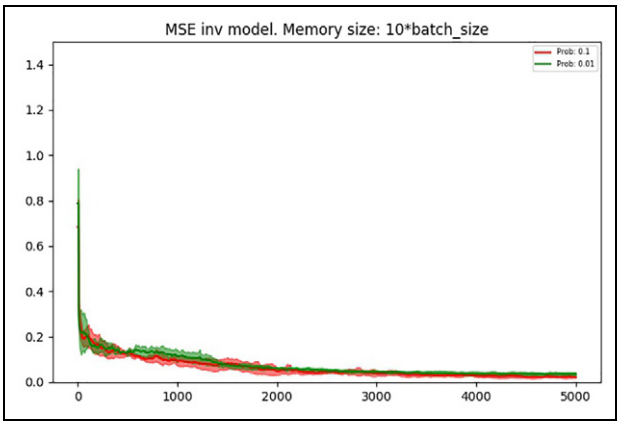


Figure 14. The mean squared error of the inverse model computed over 5000 iterations, averaged on 5 runs. The plot compares the MSE of two configurations of the system: $p_{em} = 0.1$ and $p_{em} = 0.01$. In this plot, the episodic memory size is set to 10 batches. Solid lines show the mean of the MSE over 5 runs, for each configuration. Shaded areas show the (mean − stddev, mean + stddev) areas.

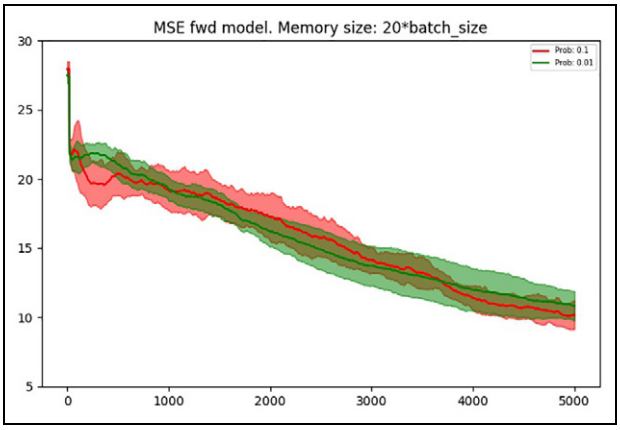


Figure 15. The mean squared error of the forward model computed over 5000 iterations, averaged on 5 runs. The plot compares the MSE of two configurations of the system: $p_{em} = 0.1$ and $p_{em} = 0.01$. In this plot, the episodic memory size is set to 20 batches. Solid lines show the mean of the MSE over 5 runs, for each configuration. Shaded areas show the (mean − stddev, mean + stddev) areas.

intrinsically motivated goal-directed exploration behaviours to the bootstrapping of motor control is in fact more evident in complex robot actuators. The goal of this work was rather to analyse the impact of episodic memories in intrinsic motivation and online deep learning on high-dimensional sensory spaces. Further work should indeed apply these algorithms in robotic platforms characterised by both high-dimensional motor and sensory spaces.

Several potential research directions are open from this study: first, the adoption of alternative predictive processes. As discussed in the introduction, more recent

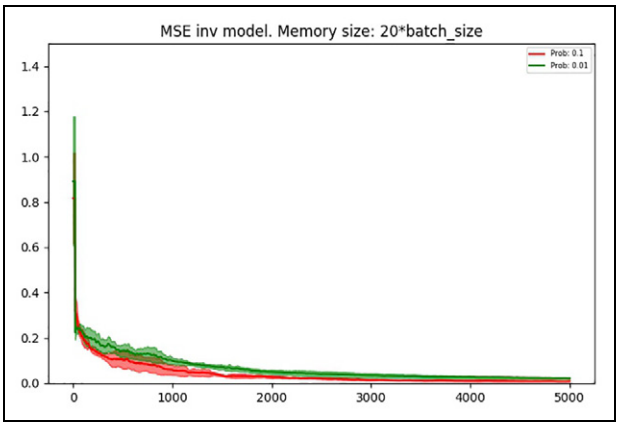


Figure 16. The mean squared error of the inverse model computed over 5000 iterations, averaged on 5 runs. The plot compares the MSE of two configurations of the system: $p_{em} = 0.1$ and $p_{em} = 0.01$. In this plot, the episodic memory size is set to 20 batches. Solid lines show the mean of the MSE over 5 runs, for each configuration. Shaded areas show the (mean − stddev, mean + stddev) areas.

approaches have been proposed in the literature, which would better leverage instrumental, goal-directed actions with epistemic, novelty-seeking, behaviour (i.e. active inference; Friston, 2010; Tschantz et al., 2019). This proposal has not been yet implemented into high-dimensional sensory spaces and robots such as the ones addressed in this study.

Second, to achieve the online learning process, image features should be learned in an online and incremental fashion. Similarly, goals could be autonomously generated through unsupervised learning techniques. We are currently exploring different possibilities, including the usage of self-organising maps, trained in an online fashion on the latent codes of the CAE. Goals would be aligned to the moving neurons of the self-organising map. Interesting insights could emerge from applying intrinsic motivation systems on dynamic goals that are autonomously generated by the learning process. More advanced goal selection strategies could be implemented, where prediction error dynamics could be analysed over longer time lapses.

Third, similar investigations should be carried out on a multimodal level. The literature on intrinsic motivation systems and goal-directed behaviours in robotics is mostly, if not totally, focusing on unimodal sensory spaces. Investigating these algorithms in the context of multimodal sensory spaces could open very interesting research directions, for instance on how to leverage the competition between modalities in the estimation of the learning progress or in the definition and selection of goals. An interesting research direction is also the integration of neuromorphic models – that is, spiking neurons – in intrinsic motivation system (see, for instance, Shi et al., 2020).

Future work on the microfarming application should consider analysing the features encoded in the 32-D vector of the CAE, once having deployed the robot on the field and having collected richer training data from the microfarming environment.

Acknowledgements

The authors would like to thank Bruno Lara and Alejandra Ciria for the very helpful feedback on the manuscript.

Author contributions

G.S. has conceived the presented ideas, implemented them and written them into the manuscript. A.P.V. has contributed to the development of the robot simulator. V.V.H. has contributed to the development of the ideas and to the revision of the manuscript. P.H., D.C. and T.W. have developed the LettuceThink robotic platform and contributed to the revision of the manuscript.

Declaration of Conflicting Interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship and/or publication of this article: G.S. has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Sk1odowska-Curie grant agreement No. 838861 (Predictive Robots). Predictive Robots is an associated project of the German DFG Priority Programme 'The Active Self'. This work has partially received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 773875 (EU-H2020 ROMI, Robotics for Microfarms).

ORCID iD

Guido Schillaci https://orcid.org/0000-0002-0975-1068

Notes

1. Please note that model updates are performed using small batches (16 samples) and the context of an episodic memory, as it will be described in the following text. We utilise here the term 'online' to express that learning is not decoupled from online behaviour.
2. For further information, please refer to https://github.com/grbl/grbl/wiki.
3. The script will be available here: https://github.com/guidoschillaci/sonylettucethink_dataset.
4. The repository containing the images will be made freely downloadable in a ZENODO.
5. The latest version can be downloaded here: https://github.com/guidoschillaci/goal_babbling_cae_episodic_memory.
6. This process is inspired by the classical idea in neuroscience about efference copy and corollary discharge (Baltieri & Buckley, 2018; Kawato, 1999; Straka et al., 2018). We are aware of more recent theories that challenged the usage of efference copies and of inverse models (Feldman, 2016; Friston, 2011; Lara et al., 2018). Future works will include reframing the predictive models of this study, getting rid of inverse models and using proprioceptive predictions to generate motor commands, as in the active inference proposal (Friston, 2010). As to the current state, we still believe that the proposed approach contributes to the state of the art in intrinsic motivation systems and online learning.
7. This may not be optimal, especially when the motor space is high-dimensional. See Tschantz et al. (2019) for more details.

References

Baldassarre, G., & Mirolli, M. (2013). *Intrinsically motivated learning in natural and artificial systems*. Springer.

Baltieri, M., & Buckley, C. L. (2018). The modularity of action and perception revisited using control theory and active inference. In T. Ikegami, N. Virgo, O. Witkowski, M. Oka, R. Suzuki, H. Iizuka (Eds.), *Artificial life conference proceedings* (pp. 121–128). MIT Press.

Baranes, A., & Oudeyer, P. Y. (2013). Active learning of inverse models with intrinsically motivated goal exploration in robots. *Robotics and Autonomous Systems*, *61*(1), 49–73.

Botvinick, M., Ritter, S., Wang, J. X., Kurth-Nelson, Z., Blundell, C., & Hassabis, D. (2019). Reinforcement learning, fast and slow. *Trends in Cognitive Sciences*. Advance online publication. https://doi.org/10.1016/j.tics.2019.02.006

Cangelosi, A., & Schlesinger, M. (2015). *Developmental robotics: From babies to robots*. MIT Press.

Cangelosi, A., & Schlesinger, M. (2018). From babies to robots: The contribution of developmental robotics to developmental psychology. *Child Development Perspectives*, *12*(3), 183–188.

Chen, C. C., & Chu, H. T. (2005). Similarity measurement between images. In *29th annual international computer software and applications conference* (*Vol. 2*., pp. 41–42). Institute of Electrical and Electronics Engineers. https://ieeexplore.ieee.org/document/1508081

Chen, S., Li, Y., & Kwok, N. M. (2011). Active vision in robotic systems: A survey of recent developments. *International Journal of Robotics Research*, *30*(11), 1343–1377.

Colas, C., Oudeyer, P. Y., Sigaud, O., Fournier, P., & Chetouani, M. (2019). Curious: Intrinsically motivated modular multi-goal reinforcement learning. In *International conference on machine learning* (pp. 1331–1340). https://arxiv.org/abs/1810.06284

Feldman, A. G. (2016). Active sensing without efference copy: Referent control of perception. *Journal of Neurophysiology*, *116*(3), 960–976.

Forestier, S., Mollard, Y., & Oudeyer, P. Y. (2017). Intrinsically motivated goal exploration processes with automatic curriculum learning. *Arxivpreprint Arxiv*. https://arxiv.org/abs/1708.02190

Frank, M., Leitner, J., Stollenga, M., Foerster, A., & Schmidhuber, J. (2014). Curiosity driven reinforcement learning

for motion planning on humanoids. *Frontiers in Neurorobotics*, *7*, Article 25.

Friston, K. (2010). The free-energy principle: A unified brain theory?*Nature Reviews Neuroscience*, *11*(2), 127–138.

Friston, K. (2011). What is optimal about motor control?-*Neuron*, *72*(3), 488–498.

Isele, D., & Cosgun, A. (2018). Selective experience replay for lifelong learning. In *Thirty-second AAAI conference on artificial intelligence*. https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/viewFile/16054/16703

Kamra, N., Gupta, U., & Liu, Y. (2018). *Deep generative dual memory network for continual learning*. https://openreview.net/forum?id = BkVsWbbAW

Kawato, M. (1999). Internal models for motor control and trajectory planning. *Current Opinion in Neurobiology*, *9*(6), 718–727.

Lara, B., Astorga, D., Mendoza-Bock, E., Pardo, M., Escobar, E., & Ciria, A. (2018). Embodied cognitive robotics and the learning of sensorimotor schemes. *Adaptive Behavior*, *26*(5), 225–238.

Lin, Z., Zhao, T., Yang, G., & Zhang, L. (2018). Episodic memory deep Q-networks. *Arxivpreprint Arxiv*. https://arxiv.org/abs/1805.07603

Masci, J., Meier, U., Cirecsan, D., & Schmidhuber, J. (2011). Stacked convolutional auto-encoders for hierarchical feature extraction. In T. Honkela, W. Duch, M. Girolami, & S. Kaski (Eds.), *Artificial neural networks and machine learning* (pp. 52–59). Springer.

Masse, N. Y., Grant, G. D., & Freedman, D. J. (2018). *Alleviating catastrophic forgetting using context-dependent gating and synaptic stabilization*. https://arxiv.org/abs/1802.01569

McClelland, J. L., McNaughton, B. L., & O'reilly, R. C. (1995). Why there are complementary learning systems in the hippocampus and neocortex: Insights from the successes and failures of connectionist models of learning and memory. *Psychological Review*, *102*(3), 419–457.

Mermillod, M., Bugaiska, A., & Bonin, P. (2013). The stability-plasticity dilemma: Investigating the continuum from catastrophic forgetting to age-limited learning effects. *Frontiers in Psychology*, *4*, Article 504.

Miranda, L., & Schillaci, G. (2019, June 16–20). An adaptive architecture for portability of greenhouse models [Conference session]. Greensys 2019 (Acta Horticulturae), *International Symposium on Advanced Technologies and Management for Innovative Greenhouses*, Angers, France.

Oudeyer, P. Y., Gottlieb, J., & Lopes, M. (2016). Intrinsic motivation, curiosity, and learning: Theory and applications in educational technologies. *Progress in Brain Research*, *229*, 257–284.

Oudeyer, P. Y., Kaplan, F., & Hafner, V. V. (2007). Intrinsic motivation systems for autonomous mental development. *IEEE Transactions on Evolutionary Computation*, *11*(2), 265–286.

Parisi, G. I., Kemker, R., Part, J. L., Kanan, C., & Wermter, S. (2019). Continual lifelong learning with neural networks: A review. *Neural Networks*, *113*, 54–71.

Parisi, G. I., Tani, J., Weber, C., & Wermter, S. (2018). Lifelong learning of spatiotemporal representations with dual-memory recurrent self-organization. *Frontiers in Neurorobotics*, *12*, Article 78.

Pathak, D., Agrawal, P., Efros, A. A., & Darrell, T. (2017). Curiosity-driven exploration by self-supervised prediction. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops* (pp. 16–17). Institute of Electrical and Electronics Engineers.

Rolf, M., & Steil, J. J. (2012, November 29). Goal babbling: A new concept for early sensorimotor exploration [Conference session]. *Humanoids 2012 Workshop on Developmental Robotics: Can Developmental Robotics Yield Human-Like Cognitive Abilities?*Osaka, Japan.

Salge, C., Glackin, C., & Polani, D. (2013). Approximation of empowerment in the continuous domain. *Advances in Complex Systems*, *16*(3), Article 1250079.

Santucci, V. G., Baldassarre, G., & Mirolli, M. (2016). Grail: A goal-discovering robotic architecture for intrinsically-motivated learning. *IEEE Transactions on Cognitive and Developmental Systems*, *8*(3), 214–231.

Schillaci, G., Hafner, V. V., & Lara, B. (2016). Exploration behaviors, body representations, and simulation processes for the development of cognition in artificial agents. *Frontiers in Robotics and AI*, *3*, Article 39.

Schlesinger, M. (2013). Investigating the origins of intrinsic motivation in human infants. In G. Baldassarre, & M. Mirolli (Eds.), *Intrinsically motivated learning in natural and artificial systems* (pp. 367–392). Springer.

Schmerling, M., Schillaci, G., & Hafner, V. V. (2015). Goal-directed learning of hand-eye coordination in a humanoid robot. In *2015 joint IEEE international conference on development and learning and epigenetic robotics (ICDL-EpiRob)* (pp. 168–175). https://doi.org/10.1109/DEVLRN.2015.7346136

Schoenberger, J. L., & Frahm, J. M. (2016). *Structure-from-motion revisited* [IEEE conference on computer vision and pattern recognition]. https://ieeexplore.ieee.org/document/7780814

Schoenberger, J. L., Zheng, E., Pollefeys, M., & Frahm, J. M. (2016). Pixelwise view selection for unstructured multi-view stereo [European Conference on Computer Vision]. https://demuc.de/papers/schoenberger2016mvs.pdf

Shi, M., Zhang, T., & Zeng, Y. (2020). A curiosity-based learning method for spiking neural networks. *Frontiers in Computational Neuroscience*, *14*, Article 7.

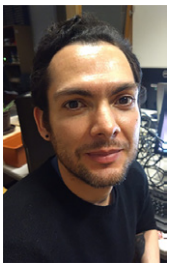
Straka, H., Simmers, J., & Chagnaud, B. P. (2018). A new perspective on predictive motor signaling. *Current Biology*, *28*(5), R232–R243.

Tschantz, A., Seth, A. K., & Buckley, C. L. (2019). Learning action-oriented models through active inference. *Biorxiv*. https://www.biorxiv.org/content/10.1101/764969v1

Zoia, S., Blason, L., D'Ottavio, G., Biancotto, M., Bulgheroni, M., & Castiello, U. (2013). The development of upper limb movements: From fetal to post-natal life. *PLOS ONE*, *8*(12), 1–9.

About the Authors

Guido Schillaci is a EU-H2020 Marie-Skłodowska Curie Fellow at the BioRobotics Institute of the Scuola Superiore Sant'Anna, Pisa, Italy. He obtained a PhD (graded *summa cum laude*) in cognitive robotics in 2014 with a EU-FP7 Marie Curie fellowship at the Humboldt-Universität zu Berlin and received a BSc and a MSc in computer engineering from the Università di Palermo, Italy. He has spent short-term academic visits at the Umeå University, Sweden, in 2012 and at the Universidad de Granada, Spain, in 2007. He has participated in several research projects funded by the EU, the German and the Italian governments. He has published ca. 40 papers in the proceedings of international scientific conferences and in academic journals. He has participated as associated or guest editor, programme committee member and reviewer in several international conferences and academic journals on different scientific topics, including cognitive and developmental robotics, neurorobotics, human–robot interaction, cognitive and developmental sciences. He has served international research funding agencies as evaluator of scientific proposals. His research interests include developmental robotics, predictive processes, internal models, robot perception and the artificial self.



Antonio Pico Villalpando was born in Chihuahua, Mexico. He received a BE in Electronics from the Instituto Tecnologico de Chihuahua where he studied from 1999 to 2004. Afterwards, he worked as an Engineer at the Museum of Science and Technology in Chihuahua. In 2008, he moved to Texcoco, Mexico and worked at the Colegio de Postgraduados developing Hardware and Software for applications in agriculture. In 2010, he obtained a scholarship from CONACyT to study a MSc degree in Computer Science at the Centro de Investigacion y de Estudios Avanzados del Instituto Politecnico Nacional, in Mexico City. He is currently doing his PhD at the Department of Computer Science at Humboldt-Universität zu Berlin, in the Adaptive Systems Group. His research interests include embedded systems, sensorimotor interaction and internal models.

Verena V Hafner is Professor of Adaptive Systems at Humboldt-Universität zu Berlin and Head of the Adaptive Systems Group at the Department of Computer Science. She holds a Master with distinction in Computer Science and AI from the University of Sussex, UK, and a PhD from the Artificial Intelligence Lab, University of Zurich, Switzerland. Before moving to Berlin, she worked as an associate researcher in the Developmental Robotics Group at Sony Computer Science Labs in Paris, France. She is part of the Programme Committee of the DFG Priority Programme The Active Self (SPP 2134), PI in the DFG Cluster of Excellence 'Science of Intelligence', PI in the graduate school on sensory computation in neural systems (DFG, 2010-2019), PI in the EU H2020 project ROMI on Robotics for Microfarms (2017–2021) and PI in the EU H2020 project HIVEOPOLIS (2019–2024). She has been reviewer for the EU, BMBF, DFG, AvH, journals and conferences. Her research interests include sensorimotor interaction and learning, joint attention, internal models and exploration strategies.

Peter Hanappe studied electronic engineering at the University of Ghent, Belgium. He wrote his PhD thesis on real-time music and sound environments at Ircam (Centre George Pompidou, Paris). When a researcher at Sony Computer Science Laboratory in Paris, he has worked on new modes of content creation and distribution that involve the participation of (online) communities. He now focuses on projects in the domain of sustainability. He was involved in setting up a collective system for monitoring noise in urban environments (NoiseTube) and in building critical components for large-scale climate simulations in collaboration with the UK Met Office. Most recently, he founded the P2P Food Lab to experiment new technologies for agroecology and community-based food systems.

David Colliaux studied mathematics, physics and cognitive science. He received his PhD in mathematics in 2011 and is now research assistant at the Sony Computer Science Laboratories Paris in the sustainability team. His research is focused on computer vision and modelling in biology.

Timothée Wintz currently works at Sony Computer Science Laboratories Paris in the sustainability team. He does research in applied mathematics, computer vision and sustainable agriculture.