A neurorobotics approach to behaviour selection based on human activity recognition

Caetano M. Ranieri1 · Renan C. Moioli2 · Patricia A. Vargas3 · Roseli A. F. Romero1

Received: 3 December 2021 / Revised: 4 August 2022 / Accepted: 14 September 2022 / Published online: 27 September 2022
© The Author(s), under exclusive licence to Springer Nature B.V. 2022

Abstract
Behaviour selection has been an active research topic for robotics, in particular in the field of human–robot interaction. For a robot to interact autonomously and effectively with humans, the coupling between techniques for human activity recognition and robot behaviour selection is of paramount importance. However, most approaches to date consist of deterministic associations between the recognised activities and the robot behaviours, neglecting the uncertainty inherent to sequential predictions in real-time applications. In this paper, we address this gap by presenting an initial neurorobotics model that embeds, in a simulated robot, computational models of parts of the mammalian brain that resembles neurophysiological aspects of the basal ganglia–thalamus–cortex (BG–T–C) circuit, coupled with human activity recognition techniques. A robotics simulation environment was developed for assessing the model, where a mobile robot accomplished tasks by using behaviour selection in accordance with the activity being performed by the inhabitant of an intelligent home. Initial results revealed that the initial neurorobotics model is advantageous, especially considering the coupling between the most accurate activity recognition approaches and the computational models of more complex animals.

Keywords Behaviour selection · Human activity recognition · Robot simulation · Neurorobotics · Bioinspired computational model

Introduction
In different branches of human–machine interaction, real-time understanding of human actions is essential for an agent to behave autonomously, proactively and effectively. This requirement has fostered research on human activity recognition (Mojarad et al. 2018).

When dealing with complex modalities (e.g., videos or data from inertial units), activity recognition approaches often rely on machine learning. For instance, video-based activity recognition has been approached by architectures based on convolutional and recurrent neural networks (Herath et al. 2017; Ma et al. 2019). Similar architectures have been proposed for inertial data, processing either raw data (Ordóñez and Roggen 2016; Garcia et al. 2019) or descriptors obtained through feature extraction methods (Steven Eyobu and Han 2018; Ashry et al. 2020).

If synchronised data from different sensors are available, activity recognition techniques may rely on multiple sensor modalities to provide more accurate results. This paradigm is known as multimodal activity recognition (Lu and
The presence of multimodal data is likely in human–robot interaction scenarios, in which social robots may act symbiotically with other pervasive devices (e.g., wearables or ambient sensors) (Amato et al. 2015; Bacciu et al. 2019).

Human–robot interaction (Selvaggio et al. 2021) and human activity recognition (Chen et al. 2021) have been fertile fields of research. Nonetheless, few approaches have been developed to link the outputs from activity recognition methods into response behaviours from a robot. Related works usually consist of direct associations between the recognised activities and the response behaviours (Georgievski et al. 2017; Li et al. 2019; Rodriguez Lera et al. 2020).

One clever way to link human activities to robot behaviours is to use a neurorobotics approach (Van Der Smagt et al. 2016). Hwu and Krichmar (2022) define neurorobotics as "the study of the interaction between neural systems and their physical embodiments in robotic platforms". Li et al. (2019) provided a comprehensive survey on the field. Robots that model structures found in the brains of living beings may be called neurorobots.

Neurorobotic models can simulate different brain systems, making them useful for several tasks (Pronin et al. 2021; Zahra et al. 2021). These models can benefit academic research from two different perspectives (Krichmar 2018; Rucci et al. 2007): (i) simulating a biological model in a neurorobot to study biological phenomena, or (ii) using biologically inspired models to address engineering problems in the field of robotics.

Our work was motivated by these two perspectives, especially the first one. We designed a neurorobotic model for behaviour selection in the context of a social robot that responded to human activities in an intelligent environment (Ranieri et al. 2021).

The proposed model embedded a computational simulation of the basal ganglia–thalamus–cortex (BG–T–C) circuit (Kumaravelu et al. 2016; Ranieri et al. 2021), a structure present in the vertebrates’ brain known to have an essential role in action selection mechanisms (Markowitz et al. 2018). Outputs from an activity recognition module were coupled with such a simulation in the form of specific stimuli. The resulting dynamics were decoded using a machine learning-based technique and converted into response behaviours for a robot. Experiments were performed in a simulated home environment.

To assess whether the proposed neurorobotic model could lead to robust behaviour selection, we introduced an alternative approach as a reference based on simple heuristics. This approach led to a deterministic behaviour selection mechanism that associated the recognised activities with robot behaviours. The rationale was to compare our approach (i.e., the neurorobotics model) with the reference presented (i.e., the heuristics approach). This comparison allowed for investigating if our model is a promising experimental platform for studies on the biological mechanisms of decision-making.

Our experiments have shown that the neurorobotics model led to outputs that could be successfully decoded into decisions for a robot. This result was more pronounced when the biological models simulated were more complex. Under certain conditions (i.e., more complex models of the BG–T–C circuit), simulations of the neurorobotics model provided a better performance than the heuristics approaches.

The remainder of this paper is organised as follows. The brain structures considered for this neurorobotics model and the computational models adopted are presented in Section "Original computational models of the BG–T–C circuit". Section "Application scenario" presents the general aspects of the robotic system and the integration between each of its modules. Section "The neurorobotics model" details the neurorobotics model. Section "Methods" depicts the methods and implementations. The corresponding results are presented in Section "Results" and discussed in Section "Discussion". Finally, the concluding remarks and directions for future research are provided in Section "Conclusion".

### Original computational models of the BG–T–C circuit

This section presents the original computational models, and basic concepts of the brain structures in the basal ganglia–thalamus–cortex (BG–T–C) circuit, responsible for motor control, decision-making, and learning (Girard et al. 2008; Liang et al. 2019; Mulcahy et al. 2020). These neural structures are characterised by competitive and complementary functions that mediate the excitation of the motor system based on inputs from the motivational system of an individual (Bariselli et al. 2019). There is also research on the potential roles of such a mechanism in robotic frameworks, including simulations in which bioinspired networks receiving different stimuli respond with different behaviours (Bahuguna et al. 2018).

The BG–T–C circuit, illustrated in Fig. 1, is formed by the motor cortex (M1), the thalamus (TH), and the basal ganglia (BG), the latter composed of a subset of structures: the striatum (Str), the globus pallidus, divided into pars interna (GPI) and pars externa (GPE), the subthalamic nucleus (STN), and the substantia nigra, divided into pars compacta (SNC) and pars reticulata (SNr). McGregor and Nelson (2019) discussed this circuit’s mechanisms and presented different theoretical models to describe it.
The pathways start with an excitatory connection from the cortex to the striatum, which projects its output neurons, named medium spiny neurons (MSN), to other structures inside the BG. In the direct pathway, the direct MSN (dMSN) inhibits the GPi, which reduces its inhibition of the TH. Then, it excites the motor cortex. In the indirect pathway, the indirect MSN (iMSN) inhibits the GPe, which reduces its inhibition to the STN, which excites the GPi. This behaviour results in inhibition of the TH and the absence of excitatory outputs to the motor cortex. In summary, the direct pathway excites the cortex (i.e., positive feedback loop), while the indirect pathway inhibits it (i.e., negative feedback loop).

In Kumaravelu et al. (2016), a computational model of the BG–T–C circuit, originally developed to study the underlying mechanisms of Parkinson’s Disease (PD), was proposed and implemented based on neural data from healthy and PD-induced (i.e., 6-OHDA lesioned) rats (Kita and Kita 2011). Eight brain structures were modelled and connected, as illustrated in Fig. 1. In particular, the direct and indirect pathways were modelled separately, representing the MSN modulation by D1 and D2 dopamine receptors in the striatum (i.e., StrD1 and StrD2, respectively). The cortex is represented by regular spiking (RS) excitatory neurons and fast-spiking (FSI) inhibitory interneurons (i.e., CtxRS and CtxFSI, respectively). A bias current was added in the TH, GPe, and GPi, accounting for the inputs not explicitly modelled. This model was designed with the ability to shift from the simulation of healthy to PD status, which is done by altering specific conductances.

Although all mammals have a similar set of BG structures that are similarly connected with thalamic and cortical structures, subtle differences between species may be found, with primates being more similar to humans than rodents (Liénard and Girard 2014; Koprich et al. 2017; Dawson et al. 2018). A data-driven approach was used in Ranieri et al. (2021) to generate a primate-based computational model of the BG-T-C circuit and the mechanisms of PD. The resulting marmoset computational model of Parkinson’s Disease was evaluated based on the differences between healthy and PD individuals, concerning the spectral signature of the brain activity (Tinkhauser et al. 2017), the dynamics of the firing rates of neurons across brain regions (van Albada and Robinson 2009), and the coherence between spike trains (Halje et al. 2019).

Application scenario

The modules of the proposed approach, and the coupling between them, are illustrated in Fig. 2. The human activities performed in a sensed environment are inferred by a machine learning algorithm (i.e., the activity recogniser). Based on the inferences performed, a behaviour selector determines the supporting behaviours for the behaviour executor. The scenario consisted of an intelligent home endowed with a social neurorobot, composing an ambient assisted living (AAL) environment (Calvaresi et al. 2017). The response behaviours corresponded to actions from the neurorobot.

Ethics implications

Before diving into the technical aspects of this work, it is crucial to consider ethical concerns related to integrated systems such as ours, which monitor a person and takes actions according to his or her behaviours. As stated in Article 12 of the Universal Declaration of Human Rights (Assembly UG 1948):

No one shall be subjected to arbitrary interference with his privacy, family, home or correspondence, nor to attacks upon his honour and reputation. Everyone has the right to the protection of the law against such interference or attacks.

This principle is valid for every person, regardless of age or medical condition. Thus, there must be an explicit agreement for anyone living in an AAL environment endowed with monitoring capabilities and proactive behaviours. Any person might be able to withdraw his or her assent at any time (Sharkey and Sharkey 2012).

Also, when deploying an intelligent environment, it is essential to provide mechanisms that ensure an acceptable level of data privacy (Könings et al. 2016). Notably, using videos as input for activity recognition in home environments may raise important privacy concerns (Caine et al. 2005). Introducing social robots in such an environment, instead of fixed cameras, may alleviate these issues since a person can send the robot away from a given...
provides a prediction vector $y_t$ at time $t$ through an activity recogniser, which classifies it into a set of predefined human activities. The output of the activity recogniser is fed to a behaviour selector, which controls a behaviour executor. The response behaviours correspond to actions from a mobile robot.

$$f_B : A \rightarrow B \cup \{b_\emptyset\} \iff f_B(a) = b,$$

$$a \in A, \quad b \in B \cup \{b_\emptyset\}$$

The robot simulation would be considered correctly completed if:

- For an activity $a$ being performed in the environment in a session $x$, the robot completed an expected response behaviour $b \in B$ before $x$ was finished; or
- No response behaviour was expected (i.e., $f_B(a) = b_\emptyset$) and the robot did not complete any of the behaviours in $B$.

It is worth noticing that, according to this evaluation policy, besides an accuracy requirement (i.e., the correct behaviour must be given in response to human activity), there was also a time constraint that must be satisfied (i.e., if required, the response behaviour must be completed while the human is still performing the given activity).

Since, by definition, $f_A(x) = a$ is not known at runtime and can only be inferred by a classifier $g \in G$ as successive prediction vectors $y'_x$ are provided, a decision-making mechanism was needed to perform adaptive decisions based on partial, time-localised predictions. In our work, we proposed a neurorobotics model that would embed the activity recognition module coupled with the bioinspired computational model in a robot (see Section “The neurorobotics model”). This model was compared to a heuristics-based approach, introduced as a reference for performance (see Section “Heuristics model implementation”).

The neurorobotics model

So far, we have formulated the problem from a general point of view. This section presents the neurorobotics model as a biologically plausible approach to address the application scenario described.

The proposed neurorobotics model determines that the behaviour selector is split into two submodules, as illustrated in Fig. 3: the bioinspired computational model (i.e., the simulated BG–T–C circuit) and a decoder. The behaviour of the bioinspired computational model is influenced by...
by the electrical stimuli based on the outputs from the activity recognition module. The decoder is responsible for processing the neural activity generated by this simulation and converting it into response behaviours for the robot. For the decoder, we employed a machine learning-based approach, inspired by developments in Brain-Computer Interfaces (BCI) (Abiri et al. 2019). On the one hand, the BG–T–C circuitry represents the decision-making structures in the brain. On the other hand, the decoder, implemented as a Convolutional Neural Network (CNN), represents the outcomes of other structures and mechanisms also present in the brain, though not explicitly modelled in our integrated system. We assume that there are structures in the brain, represented here by the CNN, capable of processing the information from the BG–T–C circuit and translating it into higher-level information (i.e., the selected behaviour).

We could have used another type of decoder. However, as shown in recent literature (Luu et al. 2021; Merk et al. 2022; Saeidi et al. 2021), machine learning and deep learning have been widely used to decode brain signals because these techniques lead to superior decoding accuracies and allow adaptation to more challenging environments.

For the bioinspired computational model, we built on the implementation used in Ranieri et al. (2021), which in turn was based on the model by Kumaravelu et al. (2016). In such an implementation (Ranieri et al. 2021), an experimental setup of either the rat-based or marmoset-based computational models was made available, allowing simulations for healthy or parkinsonian individuals. In our experiments, we considered the rat and marmoset models and performed the simulations always in the healthy state. Further details on the experimental setup will be given in Section “Methods”.

**Adapted computational models**

Motivated by the work of Mulcahy et al. (2020), we introduced two essential modifications to the computational models of the BG–T–C circuit used in this work (Kumaravelu et al. 2016; Ranieri et al. 2021). First, an additional structure, called the prefrontal cortex (PFC), was included as a variable source of excitatory stimuli towards the striatum (see Fig. 4a). Second, \( N_C = N_B \) populations of neurons were implemented as independent channels \( c \in C \), each associated to exactly one response behaviour \( b \in B \) (see Fig. 4b), as defined in Eq. 4.

\[
f_C : B \rightarrow C \iff f_C(c) = b, \quad b \in B, \quad c \in C
\]

At each timestep, the channels \( c \in C \) received a stimulus \( s \in S = \{s_1, \ldots, s_{N_B}\} \), whose intensity was based on the linear combination between a prediction vector \( y'_x \) and a weight function \( f_W \), given by Eq. 5. The actual value of \( s \) is given by function \( f_S \), defined in Eq. 6.

\[
f_W : C \times A \rightarrow \{0, 1\} \iff f_W(c, a) = \begin{cases} 1, & \text{if } f_B(a) = f_C(c) \\ 0, & \text{otherwise} \end{cases} \\
\quad c \in C, \quad a \in A
\]

\[
f_S : C \times A \rightarrow S \iff f_S(c, a) = s = \sum_{a \in A} f_W(c, a) \cdot y'_x(a) \\
\quad c \in C, \quad a \in A, \quad s \in S
\]

Considering that, as ensured by the softmax activation on the classifiers, \( \sum_{a \in A} y'_x(a) = 1 \), then \( s \in [0, 1] \mu A/cm^2 \), which has shown to be a stable, biologically plausible
interval. For a recording sequence $x$, the set of prediction vectors $y_x$ is employed to update each stimulus $s$ periodically during a corresponding computational model simulation (not to be confused with the robot simulation). For each simulation, $N_{\text{sim}}$ subsequent updates would be done for all $s \in S$, computed for the timesteps in $y_x$.

A simulation, after finished, produced a spike train for each of the brain regions modelled, contemplating all its length (i.e., all $N_{\text{sim}}$ updates were considered). The neural firing frequencies were computed according to Lánský et al. (2004), with the parameters detailed in Section “Neurorobotics model implementation”, and summed across each region of each channel. This computation resulted in $N_R \cdot N_C$ output signals for each simulation, each with length $L_U$, where $N_R = 8$ is the number of regions (see Fig. 1b).

Formally, let $u^{x,m}_x \in U$ be defined as the output for a given simulation, where $x \in X$ is a recording session, $g \in G$ is the classifier employed for activity recognition, and $m \in M$, a computational model. Therefore, let a simulation be defined as $f_U$ (see Eq. 7), whose output is as a multivariate time-series with $N_R \cdot N_C$ variables and $L_U$ timesteps.

$$f_U : X \times G \times M \rightarrow U \iff f_U(x, g, m) = u^{x,m}_x$$

$$u^{x,m}_x \in X \times G \times M, \quad u \in U$$

(7)

After the simulations were completed, the spike trains at the cortex populations were converted into temporal signals (i.e., neural firing frequencies) based on the mean firing rates across brain regions (Lánský et al. 2004). The resulting signals were segmented in smaller windows and applied to train and evaluate a convolutional neural network (CNN), which would be employed to determine the decision of the robot at each timestep of the robot simulation (i.e., the CNN-based decoder).

### CNN-based decoder

Each simulation of the computational model provided the summed neural firing frequencies of each channel and brain region, generating a data structure $u^{x,m}_x$, associated with the whole recording session that generated it. As a requirement to provide a realistic scenario for the robot simulation, time-localised decisions were required, which must be taken based only on past events.

To fulfil this requirement, each instance $u^{x,m}_x$, correspondent to the recording session $x \in X$ in the set of conditions $g \in G$ and $m \in M$ (see Eq. 7), was segmented in windows of $N_V$ timesteps, with partial superposition, producing $N_{\text{segs}}$ segments. Considering $N_X$ recording sessions in a given set of conditions, the function $f_v$ would generate a total of $N_X \cdot N_{\text{segs}}$ instances $v \in V$, as defined in Eq. 8.

$$f_v : U \times T \rightarrow V \iff f_v(u^{x,m}_x, t) = v = u^{x,m}_x[t, t + N_V]$$

$$u^{x,m}_x \in U, \quad t \in T \mid t + N_V < L_U, \quad v \in V$$

(8)

The resulting segments were employed to train the CNN-based decoder. We considered only the cortex regions to compose the input tuples for the decoder, aiming to preserve biological plausibility regarding this aspect. Given
that each channel of the computational model has two populations of cortex neurons (i.e., cortex RS and FSI) and that the experiments were performed with $N_C$ channels, associated with the response behaviours $b \in B$, the resulting instances $v$ had shape $N_V \times 2N_C$. The decoder $f_Q$ might be trained to provide a decision vector $q'_v$, which corresponds to the probability that a given segment of cortex firing frequencies, given by $v = f_c(u^{\text{cm}}_v, t)$, might be associated to a behaviour in $B \cup \{b_0\}$. This decoding function may be defined as in Eq. 9.

$$f_Q : V \rightarrow Q \iff f_Q(v) = q'_v,$$

$$\forall v \in V, \quad q'_v \in Q$$

We have adopted a one-dimensional CNN, which has shown to provide state-of-the-art results in related work (Ranieri et al. 2020) (for the architectural choices and hyperparameter settings, see Section “Neurorobotics model implementation”). Classification metrics were provided considering that the categorical output is chosen according to Eq. 10, where $d_Q$ corresponds to a response behaviour.

$$d_Q : Q \rightarrow B \iff d_Q(q'_v) = \text{argmax}(q'_v)$$

Finally, the decisions decoded would be fed to the robot actuators and turned into commands, as discussed in Section “Robot behaviours”.

**Methods**

In the neurorobotics model, as described in Section “The neurorobotics model”, the predictions from the activity recognition module were employed to stimulate a bioinspired computational model. The outputs of this model (i.e., neural firing frequencies of brain simulated regions) were decoded by a CNN-based decoder, which provided the decisions for the mobile robot. We compared our proposed neurorobotics model with a heuristics-based approach, in which two simple heuristics were proposed and evaluated to provide real-time decisions based on outputs from an activity recognition module. The heuristics approach consisted of associating the predictions of the activity recognition module to response behaviours based on simple heuristics, presented in Section “Robot behaviours”.

Figure 5 illustrates the different factors assessed in this work. We combined the heuristics approach and the neurorobotics model with two different machine learning architectures of the activity recognition module: the IMU + ambient and the video-based (see Section “Dataset and classifiers”). Two heuristics were considered and compared: the window and the exponential (see Section “Heuristics model implementation”). For the neurorobotics model, the rat and marmoset computational models were assessed (see Section “Neurorobotics model implementation”).

All code was developed in Python language. The machine learning techniques presented for the activity recognition and the CNN-based decoder were implemented with the Tensorflow/Keras framework. The computational models were implemented using the NetPyNE framework and the libraries from the NEURON simulator (Duran-Bernal et al. 2019). The robot simulation was implemented in the Gazebo simulator (Koenig and Howard 2004) with the Robot Operating System (ROS) (Quigley et al. 2009) as a middleware. The following subsections will provide the implementation details of this work.

**Dataset and classifiers**

We have adopted the HWU-USP activities dataset (Ranieri et al. 2021), a multimodal and heterogeneous dataset of human activities recorded in the Robotic Assisted Living Testbed (RALT) at Heriot-Watt University (UK). It comprises readings of ambient sensors (e.g., switches at wardrobes and drawers, presence detectors, power measurements), inertial units attached to the subjects’ waist and dominant wrist, and videos. A set of nine well-defined, pre-segmented activities of daily living was performed by the 16 participants in the data collection. A total of $N_X = 144$ recording sessions were provided, all pre-segmented and labelled (i.e., $X, A$ and $f_A$ were provided). The length of the recording sessions varied from less than 25 to over 100 s, with high variance between classes or subjects.

![Fig. 5 Factors and conditions analysed for the heuristics approach and neurorobotics model. We investigated two sets of modalities for the activity recognition module in both cases: the IMU + ambient and the video-based. For the heuristics approach, we analysed two approaches for the decision-making mechanism: window or exponential (see Section “Robot behaviours”). We considered two computational models of the BG–T–C circuit for the neurorobotics model: the rodent-based and the marmoset-based](image.png)
As the activity recognition module, we have employed the framework presented and evaluated in Ranieri et al. (2021). This framework was composed of different time-localised classifiers based on artificial neural networks, each focused on a particular modality (i.e., set of similar sensors) or set of modalities. We adopted a couple of pre-trained classifiers (i.e., the IMU + ambient and the video-based classifiers) from the framework to provide the prediction vectors, respecting the between-subjects 8-fold approach for training and evaluation. Let those classifiers be denoted by $g_{I+A} \in G$ and $g_{video} \in G$, respectively. Although both classifiers were described in Ranieri et al. (2021), we briefly present their architectures in the following paragraphs for completeness.

Classifier $g_{I+A}$ was fed with two parallel inputs: a two-seconds-long (i.e., 100 timesteps-long) time window with the raw signals from the inertial sensors and the mean values of the ambient sensors in the correspondent timestamps. A one-dimensional Convolutional Neural Network processed the inertial data (CNN) Zeiler and Fergus (2014), composed of two convolutional layers interspersed with pooling layers, followed by a Long Short-Term Memory (LSTM) recurrent layer (Hochreiter and Schmidhuber 1997), generating the feature vector $v_1$. The ambient data was processed by a single fully-connected layer, generating the feature vector $v_2$. Both $v_1$ and $v_2$ were concatenated and sent to a softmax output layer.

Classifier $g_{video}$ has taken, as input, a sequence of 25 optical flow pairs, corresponding to 2 s of video, computed with the TVL1 algorithm (Zach et al. 2007). The InceptionV3 CNN architecture (Szegedy et al. 2016) was trained to classify each optical flow pair individually. The authors adopted the CNN-LSTM architecture, which consisted of feeding each optical flow pair within a sequence to this pre-trained InceptionV3 module and feeding the resulting features as inputs to each LSTM layer’s timestep outputs, connected to a softmax output layer.

Both classifiers mentioned above were endowed with softmax activation in their outputs, which ensured that the prediction vector respects a valid probability distribution. To provide the prediction vectors, we split each recording session in $N_T = 140$ timesteps, regardless of its original length, and used the referred framework to provide the predictions on each of those timesteps. The effect is to assume that all activities have a similar length, a simplification that allowed the design of more uniform and comparable experiments related to the bioinspired computational models (Section “Adapted computational models”) and the robot simulation (Section “Robot behaviours”).

The output of the activity recognition module is, for a whole recording session $x$ processed by a classifier $g$, a total of $N_T = 144$ sets of prediction vectors $y_x = \{y_x^1, \ldots, y_x^N_T\}$, with $N_T = 140$. Outputs from both classifiers $g_{I+A}$ and $g_{video}$ were applied to all experiments, as described in the following subsections.

### Heuristics model implementation

Two policies $H$ were considered for the heuristics model, named window or exponential, that is, $H = \{h_{window}, h_{exponential}\}$. This experimental setup resulted in four conditions for evaluation in the neurorobotics model, given by the space $G \times H$.

The window policy consisted of deriving a wider prediction vector $y'_w$, correspondent to $N_{rw}$ timesteps. This was done by averaging the $N_{rw}$ most recent prediction vectors in $y_x$, from the activity recognition module, as in Eq. 11. We have set $N_{rw} = 8$, which corresponds to windows of 4 s from the recording sessions because this was the length of the segments considered for the neurorobotics model (see Section “CNN-based decoder”). On the other hand, the exponential policy consisted of deriving a prediction vector $y'_e$ that considered the whole sequence of previous prediction vectors in $y_x$, with an exponential decay across iterations, as in Eq. 12. If $R = \{r_w, r_e\}$ is the set of the functions to compute $y'_w$ and $y'_e$, then the decision $d_r$ of the heuristics approach, for either the window or exponential policies, is given by Eq. 13. It is important to note that, for the window policy of the heuristics approach, as in the neurorobotics approach, the robot can only begin to move after the first 4 s of each simulation, in which it is gathering the number of prediction vectors necessary to compute the first decision.

$$ r_w : Y \times T \rightarrow Y \iff y'_w = r_w(y_x,t) = \frac{\sum_{i=0}^{N_{rw}-1} y_x^{t-i}}{N_{rw}} \quad (11) $$

$$ r_e : Y \times T \rightarrow Y \iff y'_e = \begin{cases} r_e(y_x,t = 0) = y_x^1 \\ r_e(y_x,t > 0) = 0.9 \cdot r_e(y_x,t - 1) + y_x^t \quad y_x \in Y, \quad t \in T_X \end{cases} \quad (12) $$

$$ d_R : Y \times T \times R \rightarrow B \iff d_R(y_x,t,r) = f_R(\text{argmax}[r(y_x,t)]) \quad y_x \in Y, \quad t \in T_X, \quad r \in R \quad (13) $$

As a reference, we introduced a different approach, a condition that worked as an upper bound for performance. In this condition, the ground truth labels (i.e., the correct predictions, as expected) are directly fed to the robot simulation, providing a unique decision $d_{GT}$ every timestep, as shown by Eq. 14.
\[ d_{GR} : X \rightarrow B \iff d_{GR}(x) = f_B(f_A(x)), \quad x \in X \]

For the time constraints of our experiments, this condition would always lead to the accomplishment of the correct, expected behaviour. However, it helped evaluate the time elapsed for accomplishing the tasks in each condition analysed since it consisted of an ideal scenario without uncertainty. Hence, the tasks were accomplished in the lowest possible time.

**Neurorobotics model implementation**

Let \( M \) be the bioinspired computational model, which can be rat-based or marmoset-based, that is, \( M = \{ m_{rat}, m_{marmoset} \} \). This experimental setup resulted in four conditions for evaluation in the neurorobotics model, given by the space \( G \times M \). Each independent computational model simulation (not to be confused with the robot simulation) was run for each of the \( N_X = 144 \) recording sessions under each evaluated condition. In other words, the simulations of the computational models were required to contemplate all instances in the space \( X \times G \times M \). Hence, a total of 576 simulations of the computational model were performed.

Each of those simulations ran for 70 s with a sampling rate of 1000 Hz. The stimuli set \( S \) was updated every 0.5 s (i.e., an update frequency of 2 Hz). This led to an adaptive dynamic that would respond to successive prediction vectors \( y^t, t_{sim} \in \{ 1, \ldots, N_{Tsim} \} \), with \( N_{Tsim} = 140 \), according to the confidence of each response behaviour. The resulting spike trains in each neuron population were converted to neural firing frequencies (for details, see Lánský et al. 2004), with bins of size 20, which resulted in sequences of length \( N_U = 3500 \). As stated in Section “Dataset and classifiers”, for the experiments reported in this work, \( N_R = 2 \), hence \( N_C = 2 \). Considering that \( N_R = 8 \), the multivariate time-series \( u_t \in U \) had \( N_R \cdot N_C = 16 \) variables and \( L_U = 3500 \) timesteps, composing a data structure with dimensions \( 3500 \times 16 \).

The segments for the decoder we set to \( N_V = 200 \) timesteps (i.e., four-seconds-long) with 75\% superposition (i.e., a one-second-long step between the beginning of each segment), resulting in 66 segments. Considering the each condition was composed of \( N_v = 144 \) recording sessions, these simulations of the computational models led to a total of \( 144 \cdot 66 = 9504 \) instances \( v \in V \), for each \( (g, m) \in G \times M \).

The CNN architecture for decoding these time series into response behaviours is depicted in Table 1. It was composed of two convolutional layers, with 128 and 256 filters, respectively, interspersed with max-pooling layers.

### Table 1 Layers in the CNN-based decoder

| Layer | Type | Output shape | Free parameters |
|-------|------|--------------|----------------|
| 1     | Input | 200 × 4      | –              |
| 2     | Conv1D | 200 × 128   | 3712           |
| 3     | MaxPool1D | 100 × 128 | –              |
| 4     | Conv1D | 100 × 256   | 229,632        |
| 5     | MaxPool1D | 50 × 256  | –              |
| 6     | Global average pooling | 256 | –              |
| 7     | Softmax | 3            | –              |

The inputs to the neural network are windows of 200 timesteps from the four cortex channels of the output signals (i.e., neural firing frequencies) of the simulations under a given condition. The output is a decision vector \( q^t \) with the confidence for each response behaviour.

A global average pooling operation preceded the softmax output layer, which produced the decision vector \( q^t \).

For each set of conditions, the CNN was trained in a cross-subject 8-fold cross-validation scheme, similar to the one adopted for the activity recognition module (Ranieri et al. 2021). The input data were linearly normalised to the range [0, 1], and the classification models were trained for 40 epochs with batch size 32. The ADAM algorithm was employed, with a learning rate of \( 10^{-3} \), to optimise the categorical cross-entropy loss function. The outputs of the evaluations were stored and organised to serve as inputs to the next steps. The resulting sequences \( u_{X}^{t,m} \) were then introduced to the decision-making mechanism.

**Robot behaviours**

The behaviours \( b \in B \) consisted of transporting an object \( o \in O \), from a starting position \( z \in Z \) to a fixed destination \( z_{\text{dest}} \). This task was adopted because it comprises a basic and generic functionality for a mobile robot in a home environment. Equation 15 explains the associations between behaviours and objects. Equation 16 introduces the objects’ associations and starting positions on the map.

\[
f_O : B \rightarrow O \iff f_O(b) = o, \quad b \in B, \quad o \in O \quad (15)
\]

\[
f_Z : O \rightarrow Z \iff f_Z(o) = z, \quad o \in O, \quad z \in Z \quad (16)
\]

At each timestep \( t_{\text{robot}} \), a decision \( d \in B \cup \{ b_R \} \) (i.e., a response for each recording session \( x \in X \) of the activity recognition module) was sent to the robot simulation, composed of a mobile social robot in a home environment (for details on the platforms and implementations employed, see Section “Robot simulator”). For the neurorobotics model, this decision is given by Eq. 10, already presented in Section “CNN-based decoder”. The two policies mentioned (i.e., window and exponential) were evaluated for the heuristics approach.
The decisions were turned into commands to the robot following a table of rules, depicted in Table 2. A decision $d$ is sent to the robot at each timestep. This decision can be one of the behaviours in $b \in B$ or the "no action" behaviour $b_0$. Let $o_c$ be the object being carried by the robot at a certain timestep. Two types of situations might be considered: $d \in B$ or $d = b_0$.

The first type of situation is characterised by $d = b_0$, in which the robot must return any object that it may be carrying to the corresponding position and then stand still, waiting for any further commands. Otherwise, $d = b \in B$, the second type of situation, in which the robot is supposed to grab an object $o_c = f_O(b) \in O$ from position $f_Z[f_O(b)]$ to a destination $z_{\text{dest}}$. If the robot is not carrying any object, that is, $o_c = \emptyset$, then it must move to $f_Z[f_O(b)]$ and take the object. If it already carries the correct object, it must move towards the destination $z_{\text{dest}}$. If it is carrying the wrong object, it is, $o_c = o_k \in O$ if $o_k \neq f_O(b)$, then it must return it to $f_Z[o_k]$.

**Robot simulator**

The simulator adopted for the robotics experiments was previously made available as part of the LARa framework (Ranieri et al. 2018), consisting of a robot and a software library. The LARa robot was a mobile social robot built on top of a Pioneer P3-DX platform, endowed with a Hokuyo laser, a mini-computer, a Microsoft Kinect sensor, a microphone, and a screen, and a speaker. The LARa library was a set of functionalities implemented to control the robot based on high-level software interfaces, integrated within the Robot Operating System (ROS) (Quigley et al. 2009). Besides navigation skills and a framework for human–robot interaction, this included a platform for simulation, under conditions that resembled the actual robot’s conditions, deployed to allow offline experiments. The Gazebo simulator (Koenig and Howard 2004) was employed, and a map of a typical home environment was designed, as reproduced in Fig. 6a. The simulated robot—a simplified version of the LARa robot—is shown in Fig. 6b, while the pieces of furniture employed in the experiments are shown in Fig. 6c.

This setup comprised a realistic environment, providing several challenging aspects resembling a real-world scenario, such as sensors’ noise, communication delays and mechanical issues. The ROS platform was employed to connect this simulated environment to a navigation stack, which provided a 2D occupancy grid in which each position (i.e., cell) might be considered empty, navigable or an obstacle. This representation was generated previously to the robot simulations reported here via the GMapping algorithm (Grisetti et al. 2007) for Simultaneous Localisation and Mapping (SLAM). The mapping algorithm ran while the robot was teleoperated through the whole environment, with the laser readings and the wheels’ encoders combined to compose the occupancy grid gradually. Once the grid was created, the Augmented Monte Carlo Localisation (AMCL) and $A^*$ algorithms could be employed as a global planner to perform autonomous navigation. The navigation package was also endowed with a local planner responsible for creating adaptable short-term paths for obstacle avoidance and environmental changes.

For this work, a set of two response behaviours was defined as $B = \{b_1, b_2\}$. In Table 3, are shown the set of daily activities from the dataset (i.e., $a_p \in A$, $p \in \{1, \ldots, N_A\}$), and the expected response behaviours associated to each of those activities (i.e., $f_B(a_p)$). These were chosen respecting semantic relationships between the activities (i.e., $b_1$ is the desired response when the user is preparing meals, and $b_2$, when he is quietly consuming or exchanging information).

These behaviours were based on the assumption that the user is located in the kitchen and that sensors unaffected by the robot’s actions monitor human activities. The starting position for only the first robot simulation in a battery of experiments is given in Fig. 6. However, it had a negligible

| Decision | Object carried | Robot position | Output command |
|----------|----------------|----------------|----------------|
| $b \in B$ | $o_c = \emptyset$ | | Move towards $f_Z[f_O(b)]$ |
| $b \in B$ | $o_c = \emptyset$ | | Set $o_c = f_O(b)$ |
| $b \in B$ | $o_c = f_O(b)$ | | Move towards $z_{\text{dest}}$ |
| $b \in B$ | $o_c = o_k \in O$ if $o_k \neq f_O(b)$ | | Finish behaviour |
| $b \in B$ | $o_c = o_k \in O$ if $o_k \neq f_O(b)$ | | Move towards $f_Z[o_k]$ |
| $b_0$ | $o_c = o_k \in O$ | | Set $o_c = \emptyset$ |
| $b_0$ | $o_c = \emptyset$ | | Move towards $f_Z[o_k]$ |
| $b_0$ | $o_c = \emptyset$ | | Set $o_c = \emptyset$ |

Table 2 Table of rules associating a response behaviour $f(d) = b$ to an output command at each timestep $t_{\text{robot}}$ of the robot simulation, considering the object being carried and the current robot position.
effect on the overall results since the robot position was not reset for each simulation, as discussed later in this subsection.

As shown in Fig. 6c, we considered three pieces of furniture. These are shelf 1, associated to the robot position 
\[ z_{s1} = f_Z(o_1), \quad o_1 \in O; \]
shelf 2, associated to the robot position 
\[ z_{s2} = f_Z(o_2), \quad o_2 \in O; \]
and table, the destination, associated to the robot position \( z_{\text{dest}} \). The two specific behaviours considered for the experiments performed, \( b_1 \) and \( b_2 \), consist, respectively, of transporting object \( o_1 \) from \( z_1 \) (i.e., shelf 1) to \( z_{\text{dest}} \) (i.e., the table), and transporting object \( o_2 \) from \( z_2 \) (i.e., shelf 2) to \( z_{\text{dest}} \) (i.e., the table). Considering that shelf 2 is closer to the table than shelf 1, then the distances required for \( b_1 \) are larger than those for \( b_2 \). Consequently, the expectation was that, on average, \( b_1 \) required more time to be completed than \( b_2 \).

The maximum robot simulation time was set to \( N_{T_{\text{robot}}} = 140 \) s, with each timestep \( t_{\text{robot}} \in \{1, \ldots, N_{T_{\text{robot}}} \} \) corresponding to 1 s in the simulation. Consequently, an expected response behaviour had to finish within \( N_{T_{\text{robot}}} \) seconds to be considered complete. We configured \( N_{T_{\text{robot}}} = 140 \), which in exploratory experiments has shown to give a reasonable margin for the robot simulations.

A total of \( N_X = 144 \) robot simulations were performed for each condition analysed. The first simulation for each case began with the robot positioned as in Fig. 6c. All the subsequent simulations began without resetting the robot position after the ending of the previous one. Only the object flag, corresponding to the object \( o_c \) being carried by the robot, was cleared. Therefore, each simulation could start with the robot placed in an arbitrary position.

Results

The activity recognition module’s classification results are presented in Ranieri et al. (2021). The overall accuracy registered for the classifiers was computed by taking a set of 25 prediction vectors obtained for a recording session and averaging it. A categorical classification was provided by returning the argmax element in the averaged vector.
Following the same cross-subject partitioning adopted for evaluating the CNN-based decoder in this work, we performed a cross-validation approach. The accuracy reported for the modalities considered for the experiments reported here was 74.30% for \( g_{I+A} \), and 93.75% for \( g_{\text{video}} \).

The other modules in this work relied on necessary adaptations to frameworks previously implemented in related work, as happened to the computational models and the robot simulation, or components developed from scratch, case of the CNN-based decoder. The corresponding results are shown in the following subsections. The classification metrics from the neural firing frequencies synthesised with the bioinspired computational models are presented in Section “Simulated neural firing frequencies”. The outcomes of the robot simulations, in all conditions analysed, are presented in Section “Outcomes of the robot simulations”.

**Simulated neural firing frequencies**

A sample of the segments \( v \in V \), provided in the simulations of the computational models, is shown in Fig. 7. These simulations were generated from the rat model, stimulated according to the mechanism described in Section “Adapted computational models” taking the IMU + ambient classifier as the activity recognition module. A more significant stimulus introduced to the striatum might increase neural firing rates in the BG-T-C circuit, which might be propagated to the cortex.

The bar plot of Fig. 8 shows the overall accuracy and F1-score of the decoder, trained and evaluated according to the 8-fold cross-subject approach. The classifier used in the activity recognition module and the computational model employed are shown side-by-side.

The decoder was applied as a part of the decision-making mechanism, responsible for providing decision vectors for the robot simulation. Hence, its results might be correlated to the correct outcomes of the decisions made during the robot simulation. In other words, a good decoder accuracy might result in more correct decisions of the robot, which may more often complete the tasks with the correct outcome. The following subsection will present the experiments performed to validate this statement. These are the outcomes of the robot simulation not only for each condition but also for the different policies employed for the heuristics approach.

**Outcomes of the robot simulations**

As was mentioned before, three possible outcomes were considered for the robot simulations, with \( f_x(a) = a \) being the activity associated with a recording session \( x \in X \):

- **Correct**, if \( f_x(a) \in B \) and the activity was completed before the end of the simulation, or if \( f_x(a) = b_0 \) and no behaviour was completed;
- **Incorrect**, if the robot completed a behaviour \( b_{\text{robot}} \in B \) different from \( f_x(a) \), i.e., \( b_{\text{robot}} \neq f_x(a) \);
- **Unfinished**, if a response behaviour \( b \in B \) was expected from the robot, but no behaviour was completed before the end of the simulation.

In Section “Robot simulator”, a control condition was introduced, with ground truth decisions being sent for the robot. As expected, all robot simulations let to the correct outcome for this approach. In Fig. 9a, the outcomes for the heuristics approach are presented, with each of the policies analysed (i.e., window and exponential) being represented in different plots, each illustrating the outcomes for each classifier considered for the activity recognition module. The outcomes for the neurorobotics model are shown in Fig. 9a, with the classifiers for activity recognition (IMU +
ambient or video) and the computational models (rat or marmoset) being represented.

The times elapsed for providing the correct outcome, when a response behaviour was expected from the robot, were also recorded. Figure 10 presents the mean and standard deviations within all simulations performed for each condition. We provided two plots, separating the classifiers employed for the activity recognition module. Recognition (IMU + ambient sensors or video-based) (Ranieri et al. 2021), and the computational model considered (rat-based or marmoset-based) (Kumaravelu et al. 2016; Ranieri et al. 2021).

The ground truth approach was reproduced in both since it does not depend on prediction vectors, but on the ground truth activities.

This metric considers only the outcomes completed successfully. An approach that provides a fast response with poor accuracy would provide a low-time response, though it would not necessarily provide the correct response behaviours very often. Hence, the fact that the...
heuristics approach with the window policy led to a faster average response than the ground-truth condition is consistent. Since incorrect and unfinished outcomes were not considered for the computation of this mean value, this result only shows that, for this model, the correct outcomes were mainly associated with activities that could be completed in less time (e.g., the behaviour $b_2$).

In the results presented in Fig. 9, the numbers of incorrect and unfinished outcomes were considered separately. When computing the metrics shown in Fig. 10, incorrect and unfinished outcomes were discarded. Consequently, Fig. 10 refers to the average times elapsed only among the correct outcomes. In this scenario, we did not need to process simulations with incorrect or unfinished outcomes.

Discussion

One crucial challenge of research on neural networks is to define the architectures. In this sense, taking inspiration from brain structures may be illuminating. In this work, we took inspiration from brain structures directly related to action selection (i.e., the BG–T–C circuit). The results were superior to heuristics-based approaches based on averaging predictions, even with an initial approach based on Parkinson’s disease models. Future adaptations may be performed to design models that compete with state-of-the-art engineering solutions. Besides, biologically inspired approaches can be easily adapted through learning, which is harder to do with other approaches.

By modelling spike dynamics, we ended up with a much more biologically plausible model if compared to other approaches, such as a firing rate model. Our model allowed for controlling the parameters that affect the dynamics of the ionic flow. Since the action potentials of each neuron result from these parameters, changing them can result in many different simulated conditions. Consequently, we can modify specific parameters of the model to simulate a given condition while still controlling all of the remaining parameters. This flexibility is essential if we consider the possibilities for our neurorobotic model to help studies in neuroscience.

In our work, we could perform experiments based on a rat model designed by Kumaravelu et al. (2016) and a marmoset model derived from it using a data-driven optimisation technique (Ranieri et al. 2021). The parameters optimised included background currents, synaptic modulations and conductances. If we aimed at simulating individuals with Parkinson’s Disease, for example, we could do so by just changing a set of conductances. Similar modelling approaches apply to other medical conditions (Zheng and Kozloski 2017; Dura-Bernal et al. 2019).

Our neurorobotics model did not explicitly model mental states, as studied in some other works. For instance, Sen et al. (2020) presented an approach for classifying mental states correspondent to transition and maintenance states during bistable visual perception. Mora-Sánchez et al. (2019) relied on three datasets to study the characterisation of brain states, intending to establish correlations between the spatio-temporal properties of the physiological variables associated with these brain states and cognition.

Our system is particularly challenging, primarily because of three reasons. First, it depends on data generated by humans in an actual application scenario, which would require that volunteers are recruited to provide real-time data. Second, the behaviours might be performed by a physical robot. Third, it comprises a cascade of machine learning models: the activity recognition system, the BG–T–C simulation, and the decoder. According to this setup, a model’s input depends on another model’s output.

In other words, we implemented the proposed neurorobotics model as a sequence of individual modules, with the output of a module serving as input to the next one, according to a cascade design pattern (Lakshmanan et al. 2020). Since we were interested in the individual performances of each module, the results of one module were precomputed and stored so that they could be organised as a dataset for the next one, according to an offline evaluation strategy.

We argue that our method was valid proof of concept for validating the BG–T–C simulation as a feasible decision-
making module to produce behaviours for a mobile robot. We believe the actual deployment of such a system would be an exciting extension of this work but would contribute little to the paper’s main points.

It is essential to mention that Gazebo is a realistic simulator, which significantly reduces the reality gap, and further work is being done trying to make the simulation faster, for instance, by using GPUs and replacing HH neurons with Izhikevich neurons. Nevertheless, our proof-of-concept is an essential first step toward experiments with physical robots.

The neurorobotics model was not proposed to overcome state-of-the-art methods for robot control. Instead, the objective was to assess the feasibility of such a model that couples, for the first time, computational models of the BG–T–C circuit with real-time activity recognition to drive behaviour selection for an autonomous robot in an application scenario. We believe that this initial model could lead to a platform that can inform neuroscientists and help test hypotheses before moving to animal models, therefore providing a tool for neuroscience research (Pimentel et al. 2021).

Regarding the decoder, although CNNs are primarily inspired by visual processing mechanisms in the brain (Haykin 2008), it is sensible to consider that the underlying principles of CNNs are not restricted to visual processing but rather a structure that could be related to other aspects of animal sensory encoding and decoding (Drakopoulos et al. 2021; van der Heijden and Mehrkanoon 2022). Thus, we assume that CNNs are a valid approach for modelling brain mechanisms. On top of that, CNNs excel in image and pattern recognition and other classification tasks (Li et al. 2021). This is the main reason why we opted for this approach as a neural decoder in our model.

The results from the CNN-based decoder, shown in Fig. 8, confirmed some expectations regarding the output signals produced by the simulations of the computational models according to the stimuli provided: it performed better for the video-based classifier than for the IMU + ambient, and for the marmoset-based model, compared to the rat-based. The accuracy and F1-score metrics were very close, which, considering a strictly balanced dataset, suggested that the results were not affected by any serious issues regarding the trade-off between precision and recall.

All evaluations led to an accuracy measure of over 70% for three classes (i.e., response behaviours $b_1$, $b_2$ or $b_0$). It is important to consider that the stimuli came from noisy prediction vectors from activity recognition algorithms, whose accuracy is variable across successive segments (Ranieri et al. 2021), with overall accuracy values of 74.30%, for $g_{14}$, and 93.75%, for $g_{15}$. These results show that the neural activity provided by the computational models could be reliably interpreted by the proposed decoder, even considering segments of limited length (i.e., four-second-long segments within a 70-seconds-long sequence). Hence, this particular technique for brain signals, analysed in previous studies for processing related neuronal data of the BG–T–C circuit (Oh et al. 2018; Ranieri et al. 2020), has shown to be suitable for the neurorobotics model proposed.

Since the accuracy measure of the classifier for activity recognition was significantly higher for the video-based classifier than for the IMU + ambient, it was expected that it could be more easily decoded by the neural network, which was confirmed by the decoder results (see Fig. 8). Also, the marmoset-based model led to better decoding performances than the rat-based model, which also meets the expectations, considering a more sophisticated morphology and dynamics in the underlying brain structures in primates than in rodents (Liénard and Girard 2014).

A conceptual aspect worth mentioning is that the experiments considered a small, fixed set composed of three behaviours, one of which is a “no action” behaviour in which no action was expected from the robot, as described in Section “Robot behaviours”. Although the neurorobotics model presented in Section “The neurorobotics model” could theoretically address any number of robot behaviours, practical aspects might be considered.

The bioinspired computational model outputs a high-dimensional, multivariate time series in which each variable corresponds to one population of neurons simulated. Since the neurorobotics model design requires each behaviour to be associated with a channel containing a population of neurons for every region of the BG–T–C circuit, the number of variables in this time series increases linearly with the number of behaviours. As a result, the CNN-based decoder responsible for interpreting this time series may suffer from the course of dimensionality if the number of behaviours is too high (Wojtowytsch and Weinan 2020). In this context, prohibitively large amount of data could be needed for the system to behave appropriately. Hence, additional research would be necessary for dealing with the high dimensionality of the time series consisting of mean firing frequencies.

Another limitation is that the requirement for a population of neurons associated with each robot behaviour implies that the cost of the simulations of the computational model increases as more behaviours are included. Although this situation has little effect when a small number of behaviours are added, it could lead to significantly degraded performance for a large number of behaviours. Therefore, although we provided a formulation of our method so that any number of behaviours may be introduced, the current state of development of the
neurorobotics model does not allow for an arbitrarily large set of behaviours.

Regarding the robot simulations, heuristics approaches were evaluated parallel to the neurorobotics model. In most experiments performed in this work, better performances were found for the models fed by the video-based classifier than those fed by the IMU + ambient classifier, which was expected, since the video classifier is expressively more accurate (Ranieri et al. 2021). As shown in Fig. 9a, the window policy led to a lower number of correctly completed response behaviours, especially when fed with prediction vectors from the IMU + ambient classifier (less accurate). This condition may be the fairest comparison to the neurorobotics model since it limits its decisions to data from the four-second-long segment that precedes a given decision, the same constraint applied to the CNN-based decoder.

In this context, the neurorobotics model provided more accurate outcomes in most conditions, especially for the marmoset model. For the IMU+ambient modality of the activity recogniser, the window policy of the heuristics approach led to 79.9% of correct outcomes. This result was surpassed by the 84.7% result for either the rat or marmoset models. For the video modality, the window policy of the heuristics approach led to 86.0% of correct outcomes, which was only slightly above the rat model, which hit 85.3%, and expressively below the marmoset model, which hit 93.7%. These results point out that the proposed neurorobotics model, in the conditions analysed in this study, may lead to better outcomes than simple heuristics for a real-time task of an autonomous robot.

This result must be understood given the objectives of this study, which was to propose a neurorobotics model that embeds a simulation of the BG–T–C circuit, which has multiple functions within the mammals’ brain, coupled with an activity recognition module, to assess its feasibility in providing a decision-making system for a mobile robot. The fact that it performed better than heuristics-based approaches indicates that the neurorobotics model could drive the robot to correct behaviour selection beyond what could be achieved by simply averaging the outputs of the activity recognition module in a deterministic manner. Nevertheless, a viable module based on the BG–T–C circuit, which could be a feasible, advantageous solution for addressing engineering problems, would still need an enhanced design.

A particularity was found for the exponential policy of the heuristics approach: it led to similar results for either the video and IMU + ambient conditions (i.e., 88.2% and 88.8% of correct outcomes, respectively), both with more correct outcomes than those of the window policy. This result is relevant since it reveals that, by performing a long-term aggregation of prediction vectors obtained from a single recording session, it may be possible to compensate for lower accuracy values provided by specific classifiers that work with different sets of sensors. This possibility might be considered in practical applications, in which more informative modalities that usually lead to high accuracies, such as videos, may be either difficult to be obtained due to privacy concerns (Fernandes Junior et al. 2016) or unfeasible to provide real-time outputs due to the high computational cost inherent to the operations required for processing them (Rodríguez-Moreno et al. 2019).

Regarding the different conditions considered for the neurorobotics model (i.e., which architecture was used for the activity recogniser, and which bioinspired computational model was employed), the expectation was that, when applied to the robot simulation, the number of correct outcomes would be comparatively proportional to the accuracy measures of the decoder (see Fig. 8). As shown in Fig. 9b, this expectation was met for most conditions, although some exceptions were found.

Better results for the marmoset model were expected since the number of neurons and the connectivity are larger (Prescott et al. 2006; Koprich et al. 2017). The results of the decoder, previously discussed, corroborate this hypothesis. For the robot simulations, considering the video modality, the marmoset model led to the best results found among all of the simulations, with 93.7% of correct outcomes, against 85.3% achieved by the rat model. However, the IMU + ambient modality results were similar for both models. A possible explanation for this result is that such an increased capacity could compensate for the mistakes for more accurate activity recognition. In other words, the prediction vectors across successive segments could assign higher confidence values (i.e., probabilities) to the expected label (i.e., the ground-truth activity) for the video-based classifier than for the IMU + ambient. The more sophisticated marmoset-based model took more advantage of it than the rat-based model.

An interesting observation is that, for the neurorobotics model, the accuracy of the CNN-based decoder was lower than the proportion of correct outcomes from the robot simulation. There are a couple of factors that contribute to this result.

First, the behaviour selection mechanism integrates the confidence scores for each output class in all scenarios evaluated. This approach differs from assigning a single class with the highest probability, which is how accuracy assessment is performed. Integrating over the confidence scores helps attenuate the effect of wrong classifications with low confidence (e.g., for a given prediction vector, the class with the highest probability may provide slightly higher confidence than some other class).

Second, the robot must perform a set of actions within a given time constraint to accomplish a task in the simulated
environment. If the correct predictions are given consistently for a sufficiently large sequence within a session, the correct behaviour may be completed even if the accuracy score is lower at other sequences in the session. In other words, the overall accuracy computed for the decoder is not uniformly distributed throughout an activity session.

In Ranieri et al. (2021), the authors analysed the confidence for the ground truth labels obtained from the model throughout the sessions. For some activities, the confidence was higher at the beginning of the sessions and lowered for the remainder. Such a result favours an inherently sequential analysis such as ours, in which correct predictions at the beginning allow for accomplishing the correct behaviour before the wrong predictions significantly affect the outcomes.

One of the possible outcomes is that the robot does not complete any of the tasks, which is the desired outcome for a third of the activities considered. This outcome may be achieved if these classes are decoded successfully, but also for any situation where the behaviour selection mechanism does not systematically and consistently output a specific, wrong decision.

We can see only modest variations across conditions by measuring the time elapsed in the robot simulations with correct outcomes. An important observation regarding this metric is that a fast response is not necessarily an indication of good performance since this result is affected not only by the assertiveness of the correct outcomes (i.e., few changes of a decision within a simulation) but also by the accuracy of the simulations in a given set of conditions. For instance, a given condition may lead to a fast response when it provides the correct outcome, but most simulations may lead to an incorrect or unfinished outcome.

For the neurorobotics model, the times were approximately similar between both classifiers, except for the marmoset model, which took significantly longer to finish, on average, when fed with the video-based classifier. For the heuristics approach, the video-based classifier led to longer times for completing the behaviours, probably because some of the changes in decisions (i.e., the robot is performing behaviour \( b_1 \), but the decision-making mechanism changes it to \( b_2 \) after receiving new, updated prediction vectors) within the simulations allowed for completing more simulations with the correct outcome. The same reason explains why the correct outcomes of the window policy for the IMU + ambient classifier led to a faster response, on average than the ground-truth value.

As mentioned in the last paragraphs of Section “Outcomes of the robot simulations”, the computation of the average times elapsed considered only the simulations in which the correct behaviours were completed (i.e., correct outcomes). Simulations in which an incorrect behaviour was completed (i.e., incorrect outcomes) or no behaviour was completed (i.e., unfinished outcomes) were not assessed for computing this metric. Therefore, increments or decrements in the number of incorrect or unfinished actions could not influence it directly.

Analysing the results presented in Figs. 9 and 10, stratified by the type of input (i.e., IMU + ambient or video), a correlation can be established between the proportion of correct actions and the average times elapsed.

For the IMU + ambient models, ordering from the lowest to the highest number of correct actions, we get the following sequence: heuristics-window, neurorobotics-marmoset and neurorobotics-rat (tied), heuristics-exponential. Similarly, the ordered times elapsed, from the lowest to the highest, were: heuristics-window, neurorobotics-marmoset, neurorobotics-rat, and heuristics-exponential. For the video-based models, the same type of correlation was observed. Here, the neurorobotics-rat model performed less accurate and faster than the other models, and the neurorobotics-marmoset model performed the best and the slowest.

One explanation for such a correlation is that the most challenging decisions accomplished by the most accurate models took longer to be finished. The less accurate models could not finish these actions. Hence they were not included in the computation of the average times, which considers only the correct outcomes. Thus, the average time elapsed for completing the correct outcomes is reduced in these models since they refer primarily to sequences completed with less uncertainty.

### Conclusions and future work

In this paper, we proposed a neurorobotics model comprised of computational models of brain structures coupled with activity recognition techniques for creating a decision-making mechanism to provide effective behaviour selection to a mobile robot in a simulated environment.

The chosen application scenario was a simulated smart home or an ambient assisted living (AAL) place, where data from the sensed environment was processed with a previously designed activity recognition bioinspired framework. As a proof of concept, the initial neurorobotics model was compared to a heuristics approach, created to provide real-time decisions based on the outputs from an activity recognition classifier.

The neurorobotics model used computational models (CM) of the basal ganglia-thalamus-cortex (BG–T–C) circuit, initially designed to study Parkinson’s disease’s underlying mechanisms. We modified the CM so that the outputs from the activity recognition module stimulated the striatum of the circuit. A Convolutional Neural Network (CNN) decoded the resulting cortical spike activity to
provide decisions to the robot. Different conditions were analysed, including whether the computational models were based on rodent or primate models.

We reported the results concerning the decoding accuracy obtained in each condition for the computational model and the outcomes of the robot simulations, considering the neurorobotics model and the heuristics approach. The expectations were met for most of the different conditions regarding the neurorobotics model, with the primate-based computational model leading the best outcomes among the simulations analysed.

Hence, one can conclude that the proposed neurorobotics model is promising as an embedded tool for understanding the neurophysiological aspects of animal behaviour and as a practical component to integrate decision-making mechanisms for behaviour selection in mobile robots engaged in human–robot-interaction scenarios.

We performed the experiments with a limited set of three behaviours. Although we provided a formulation of our model so that any number of behaviours may be introduced, in practice, this number was not too large for the module that comprises the computational model, as each behaviour requires a population of neurons to be introduced. The decision-making mechanism might need improvements for a large set of behaviours or for a behaviour space that is not categorical, which could be an exciting follow-up to this work.

Further investigations may include a robot in an actual physical environment where human participants may perform activities. These investigations would require integrating the different modules shown in the presented pipelines, thus ensuring that all of them can work in real-time. Such an experiment may validate our neurorobotics model in even more challenging conditions and scenarios, which may foster a wide range of applications.

Author Contributions
All authors contributed to the design of the experiments. Ranieri provided the specific methods and implementations, performed the experiments and analysed the results. Moioli, Vargas and Romero revised the methods and results presented, contributing to the discussion. Ranieri wrote the draft of the paper, revised by the other authors.

Funding
This work was funded by the Sao Paulo Research Foundation (FAPESP), grants 2017/02377-5, 2017/01687-0, 2018/25902-0 and 2021/10921-2, and the Neuro4PD project - Royal Society and Newton Fund (NAF/ R2/180773). Moioli acknowledge the support from the Brazilian institutions: INCT INCMAQ of the CNPq/MCTI, FAPERN, CAPES, FINEP, and MEC. This research was carried out using the computational resources from the CeMEAI funded by FAPESP, grant 2013/07375-0. Additional resources were provided by the Robotics Lab within the ECR, and by the Nvidia Grants program.

Data availability
The HWU-USP activities dataset is available at Data Dryad (Ranieri et al. 2021). All code employed in this paper is available at Github, under the following repositories: Activity recognition framework: https://github.com/cmranieri/Deep-Activity-Recognition, Bioinspired computational model and decoder: https://github.com/cmranieri/Bioinspired-behaviour, Robot simulation: https://github.com/cmranieri/robot-simulation

Declarations

Conflict of interest
The authors declare that they have no conflict of interest.

Consent for publication
All authors agreed to publish this paper in its present form.

References
Abiri R, Borhani S, Sellers EW, Jiang Y, Zhao X (2019) A comprehensive review of EEG-based brain-computer interface paradigms. J Neural Eng 16(1):011001
Amato G, Bacciu D, Broxvall M, Chessa S, Coleman S, Di Rocco M, Dragne M, Gallicchio C, Gennaro C, Lozano H, McGinnity TM, Micheli A, Ray AK, Renteria A, Saffiotti A, Swords D, Vairo C, Vance P (2015) Robotic ubiquitous cognitive ecology for smart homes. J Intell Robot Syst Theory Appl. https://doi.org/10.1007/s10866-015-9978-6
Ashry S, Ogawa T, Gomaaw W (2020) CHARM-deep: continuous human activity recognition model based on deep neural network using IMU sensors of smartwatch. IEEE Sens J 20(15):8757–8770. https://doi.org/10.1109/JSEN.2020.2985374
Assembly UG (1948) Universal declaration of human rights. UN General Assembly
Bacciu D, Di Rocco M, Dragne M, Gallicchio C, Micheli A, Saffiotti A (2019) An ambient intelligence approach for learning in smart robotic environments. Comput Intell. https://doi.org/10.1111/coin.12233
Bahuguna J, Weidel P, Morrison A (2018) Exploring the role of striatal D1 and D2 medium spiny neurons in action selection using a virtual robotic framework. Eur J Neurosci 49(6):737–753. https://doi.org/10.1111/ejn.14021
Bariselli S, Bobbs WC, Creed MC, Kravitz AV (2019) A competitive model for striatal action selection. Brain Res. https://doi.org/10.1016/j.brainres.2018.10.009
Caine KE, Rogers WA, Fisk AD (2005) Privacy perceptions of an aware home with visual sensing devices. Proc Hum Factors Ergon Soc Annu Meet 49(21):1856–1858. https://doi.org/10.1177/154193120504902108
Calvaresi D, Cesarini D, Sernani P, Marinoni M, Dragoni AF, Sturm A (2017) Exploring the ambient assisted living domain: a systematic review. J Ambient Intell Humaniz Comput 8(2):239–257
Chen K, Zhang D, Yao L, Guo B, Yu Z, Liu Y (2021) Deep learning for sensor-based human activity recognition: overview, challenges, and opportunities. ACM Comput Surv (CSUR) 54(4):1–40
Dawson TM, Golde TE, Lagier-Tourenne C (2018) Animal models of neurodegenerative diseases. Nat Neurosci 21(10):1370–1379. https://doi.org/10.1038/s41593-018-0236-8
Drakopoulos F, Baby D, Verhulst S (2021) A convolutional neural-network framework for modelling auditory sensory cells and synapses. Commun Biol 4(1):1–17
Dura-Bernal S, Suter BA, Gleeson P, Cantarelli M, Quintana A, Rodriguez F, Kedziora DJ, Chadderdon GL, Kerr CC, Neymotin SA, McDougal RA, Hines M, Shepherd GM, Lytton WW (2019)
Mulkahy G, Atwood B, Kuznetsov A (2020) Basal ganglia role in learning rewarded actions and executing previously learned choices: healthy and diseased states. PLoS ONE 15(2):1–26

Oh SL, Hagiwara Y, Raghavendra U, Yuvaraj R, Arunkumar N, Murugappan M, Acharya UR (2018) A deep learning approach for Parkinson’s disease diagnosis from EEG signals. Neural Comput Appl. https://doi.org/10.1007/s00521-018-3689-5

Ordóñez F, Roggen D (2016) Deep convolutional and LSTM recurrent neural networks for multimodal wearable activity recognition. Sensors 16(1):115. https://doi.org/10.3390/s16010115

Pimentel JM, Moioli RC, De Araujo MF, Ranieri CM, Romero RA, Broz F, Vargas PA (2021) Neuro4pd: an initial neurorobotics model of Parkinson’s disease. Front Neurorobotics 88

Prescott TJ, Montes González FM, Gurney K, Humphries MD, Pimentel JM, Moioli RC, De Araujo MF, Ranieri CM, Romero RA, Ranieri CM, MacLeod S, Dragone M, Vargas PA, Romero RA (2021) Unveiling Parkinson’s disease dysfunction phenotypes. Front Comput Neurosci 11:70

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensors hold exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.