# Behavioral Imitation with Artificial Neural Networks Leads to Personalized Models of Brain Dynamics During Videogame Play

**Anirudha Kemtur**[1,2,3*], **Francois Paugam**[1,2,3], **Basile Pinsard**[3], **Pravish Sainath**[1,2,3], **Maximilien Le Clei**[3], **Julie Boyle**[3], **Karim Jerbi**[1,2,3,4], **Pierre Bellec**[1,3,4]

***For correspondence:** anirudha.kemtur@gmail.com (Anirudha Kemtur)

[1]Computer Science Dept., Université de Montréal; [2]Mila – Quebec AI Institute; [3]Centre de Recherche de l'Institut Universitaire de Gériatrie de Montréal; [4]Psychology Dept., Université de Montréal

**Abstract** Artificial Neural networks (ANN) trained on complex tasks are increasingly used in neuroscience to model brain dynamics, a process called brain encoding. Videogames have been extensively studied in the field of artificial intelligence, but have hardly been used yet for brain encoding. Videogames provide a promising framework to understand brain activity in a rich, engaging, and active environment. A major challenge raised by complex videogames is that individual behavior is highly variable across subjects, and we hypothesized that ANNs need to account for subject-specific behavior in order to properly capture brain dynamics. In this study, we used ANNs to model functional magnetic resonance imaging (fMRI) and behavioral gameplay data, both collected while subjects played the Shinobi III videogame. Using imitation learning, we trained an ANN to play the game while closely replicating the unique gameplay style of individual participants. We found that hidden layers of our imitation learning model successfully encoded task-relevant neural representations, and predicted individual brain dynamics with higher accuracy than models trained on other subjects' gameplay or control models. The highest correlations between layer activations and brain signals were observed in biologically plausible brain areas, i.e. somatosensory, attention, and visual networks. Our results highlight the potential of combining imitation learning, brain imaging, and videogames to uncover idiosyncratic aspects of behavior and how those relate with individual brain activity.

## Introduction

Artificial Neural Networks (ANNs) trained with deep learning have emerged as good encoding models of brain activity in varied cognitive domains. In a brain encoding framework, a computational model is first trained to do a task of interest using a rich sensory data stream as input, and then the internal states of that model are used to predict the brain responses while performing the same task (***Schrimpf et al., 2018***). Brain encoding with ANNs has been applied in various cognitive domains such as visual object recognition (***DiCarlo et al., 2012***), audition (***Kell et al., 2018***), and natural language processing (***Caucheteux and King, 2022***). However, previous applications of ANNs have focused mainly on passive sensory processing and much less on active tasks involving decision making.

Agency in the environment is a key aspect of human behavior, which needs to be incorporated

in this nascent field at the intersection of neuroscience and artificial intelligence (***Zador et al., 2022***). Videogames strongly engage participants' attention, emotions, motor and decision making abilities (***Anderson et al., 2011***; ***Bavelier et al., 2012***) therefor offering a rich framework to model the integration of brain dynamics across the perception to action loop (***Bellec and Boyle, 2019***).

A recent work demonstrated that ANNs can successfully be applied to brain encoding in videogames, specifically for classic Atari games (***Cross et al., 2021***). However, Cross and colleagues relied on a generic reinforcement learning model whose behavior was not specifically tuned to subjects' individual gameplay. Although this generic model approach may have been adequate for relatively simple games, it does not leave much room for exploring inter-individual behavioral variations during videogame play. Of particular interest is the fact that individual human videogame play can be rich and diverse given the complex environments. In this work, we aimed to test the hypothesis that ANNs trained to closely imitate individual gameplay would translate to better brain encoding models for complex video game environments (see Figure 1 for an outline of the approach).

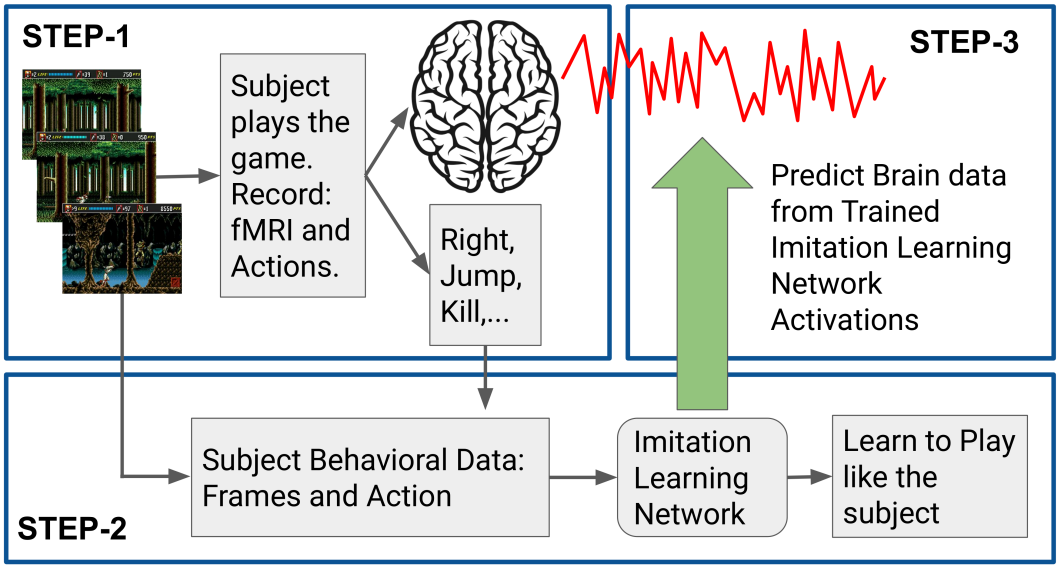


**Figure 1.** Experiment Pipeline. **Step-1**: Record videogame frames, fMRI data, and key presses of the subject. **Step-2**: Train an imitation learning neural network using frames and action data, such that it learns to play with a gameplay style similar to the subject. **Step-3**: Use activations from the neural network model to linearly predict the fMRI data.

## Results

### ANNs trained to imitate individual gameplay successfully encoded brain activity

#### Behavioral results

We first aimed to train ANNs that could imitate the behavioral gameplay of individual subjects. We trained a deep ANN composed of vision (CNN) and recurrent (LSTM) layers using behavioral cloning, to predict button presses directly from pixel-level video frames from the game (out of 128 possible actions)(Figure 5). Leave-one-session-out cross-validation was performed, and the model imitated the subjects' actions with an accuracy of 62-68%; sub-01: 62%, sub-02: 64%, sub-04: 68%, sub-06: 66% (Chance level: 0.8%). To complement these quantitative performance metrics, we also used trained models to play the game for entire levels, by feeding the actions predicted by the model back into the game emulator. Qualitatively we found that trained models performed adequately, being able to progress through levels and often complete them, and taking sequences of actions

analogous to individual subjects' gameplay. A demo video of imitation learning model playing the game with a subject-specific style is available at this Link.

## Brain encoding

Next, we tested whether the representations learned by individual imitation models could be used to predict task-related fluctuations in brain activity, for the same subjects. We implemented a brain encoding analysis, where the video frames of the game of a player were fed into the imitation model, the activity of layers of the model was used as input to a Ridge regression that predicted brain activity in a series of functional parcels (Figure 6).

The coefficient of determination $R^2$ was calculated between the predicted fMRI data from the ridge regression model and the actual fMRI data for each parcel (Figure 2). We saw that the imitation learning model predicted brain dynamics with high accuracy in distributed brain regions. Both the quality of brain encoding and the regions with highest encoding scores were largely consistent across subjects. Predicted brain regions included the visual cortex (in particular the dorsal stream), sensorimotor cortex, as well as some frontoparietal areas, and these results were comparable to those obtained by Cross and colleagues ***Cross et al.*** (***2020***). We also found that encoding $R^2$ were poor in subcortical regions for all subjects, which likely reflects the low signal-to-noise ratio(SNR) in those regions that is common with multi-band accelerated fMRI ***Srirangarajan et al.*** (***2021***). Although these brain regions were consistent across subjects, the average and maximum $R^2$ varied markedly across subjects. These results likely reflect systematic differences in data quality across subjects and notably varied levels of motion. Note that noise ceiling could not be estimated for this task as gameplay varied substantially across repetitions and subjects.

Here we presented the encoding results from the trained imitation learning model where features from all the layers were used to predict the brain data. We also analyze what effect the early layers vs later layers of the model have on brain encoding accuracy as well as which brain regions each of them help to encode better (Appendix 1).

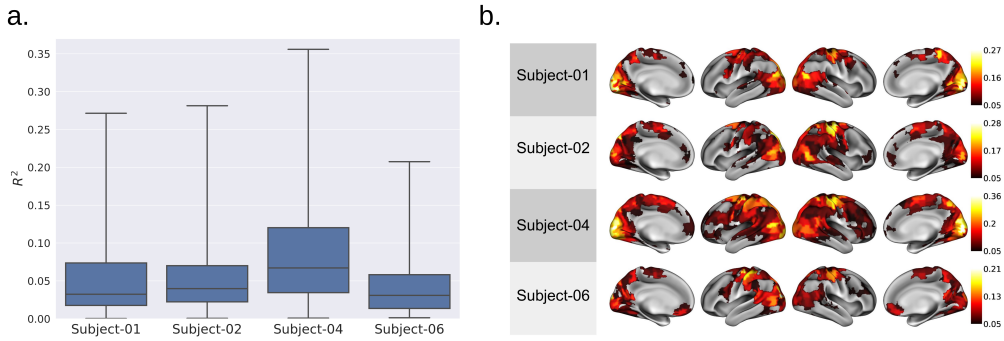


**Figure 2.**
Percentage of variance explained in actual fMRI BOLD data by responses predicted by the model. The fMRI data were predicted from the internal representations of separate imitation learning models, each trained on the corresponding subject's gameplay. Panel **a**: Distribution of brain encoding $R^2$ (percentage of variance explained in actual fMRI BOLD data by responses predicted by the model) shown across four subjects. For each subject distribution of $R^2$ across 444 parcels is plotted using a box plot showing the minimum, first quartile, median, third quartile, and maximum. Panel **b**: $R^2$ across four subjects plotted on a brain map.

## Subject-specific imitation translates into better brain encoding than models trained to imitate other subjects' gameplay

### Behavioral results

In the previous section, we saw that the imitation learning model was able to predict behavioural actions with an accuracy of up to 65%. Here, we tested whether these imitation models were subject-specific.Specifically, we evaluated if the imitation learning model trained on a subject's gameplay data can model the behavior of that same subject better than a model that knows to play the game but in a different gameplay style, drawn from the other three subjects. The results showing the within and between subject comparison are shown in Figure 3**a**. While the imitation learning model trained on the subject own gameplay was able to replicate the gameplay data up to 65% accuracy, the models trained on a different gameplay style only achieved 35% accuracy. This result was observed consistently across all four subjects.

### Brain encoding

Next, we wanted to see if the subject-specificity of imitation models translated into better brain encoding (higher $R^2$). Specifically, we tested if a subject-specific imitation model was able to encode the brain data of that target individual better than imitation learning models trained on a different, alternative subject gameplay. This alternative subject was selected as the one most behaviorally similar to the target, i.e. whose behavioral model reached highest accuracy on the target subject (sub-01 was paired with sub-06, sub-02 with sub-01, sub-03 with sub-01 and sub-04 with sub-02). We passed prerecorded gameplay frames of the target subject into these two models: (1) the imitation model trained on the target subject and (2) the imitation model trained on the the best behavioural alternative subject. The internal activations from these models were used to predict brain data of the target subject using ridge regression.

Distribution of $R^2$ across the parcels for each subject-specific model and the best behavioural alternative model were plotted (Figure 3**b**). Across all four subjects, we observed that the target (same-subject) imitation model was better at encoding the brain data than the alternative model selected for that subject. These differences were significant at $q < 0.05$ (Wilcoxon signed-rank test with false discovery rate correction for multiple comparisons across parcels). Same-subject imitation learning models better encode brain data than their corresponding alternative subject model in $6\%$, $35\%$, $37\%$, $26\%$ of parcels in Subjects 01, 02, 04 and 06 respectively.

Figure 3**c** shows in which brain regions the subject-specific model encoded the brain dynamics better than the best behavioural alternate subject model. Specifically, we found improved encoding in motor, somatosensory and visual cortices, compared to models trained with different gameplay. These areas are key to processing the input of the task (visual stimuli) and generating outputs (motor command). While imitation models trained on different gameplay knew how to play the game, the actions they might take were often dissimilar to that of the target subject, as demonstrated by the loss in accuracy for predicting actions. Representations of these subject-specific actions likely explain the ability of subject-specific imitation models to captures more brain variance in motor and somatosensory areas.

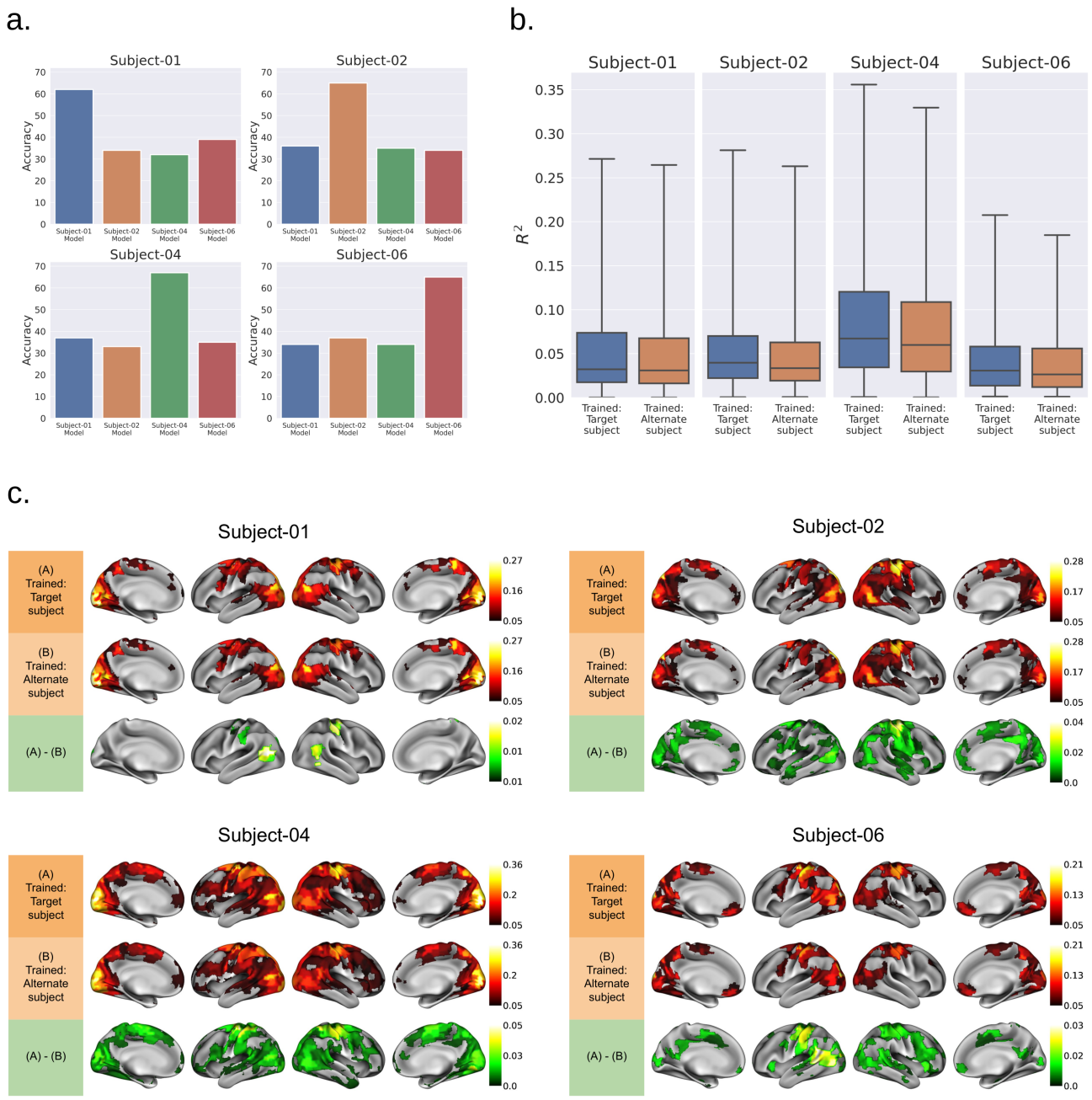


**Figure 3.**
Panel **a**: Behavioral imitation learning accuracy across four subjects. X-axis showing which subject the model was trained on.
Panel **b**: Distribution of brain $R^2$ comparing encoding performed using model trained on target(same) subject vs alternate(best out of rest three) subject. Panel **c**: $R^2$ values on a brain map shown for each of the four subjects. **First row**: Brain map of $R^2$ from the trained model (same subject gameplay). **Second row**: Brain map of $R^2$ from the Trained model (Different gameplay). **Third row**: Brain map of $R^2$ (trained model: same subject gameplay) - $R^2$ (trained model: different gameplay) , where the $R^2$ values are subtracted parcel wise and plotted (only parcels where the difference is significant at q<0.05 are colored).

## Subject-specific imitation models outperformed a series of control models for brain encoding

We compared brain encoding with subject-specific imitation models to a series of control models: (1) PCA features derived from gameplay input pixel (Pixel-space model); (2) one-hot encoding of the controller button presses (Action-space model); and, (3) passing the subjects' gameplay frames to a CNN-LSTM model with random weights (Untrained model).

Distribution of $R^2$ across the parcels for imitation and control models were plotted (Figure 4). Across all four subjects, the subject-specific imitation model was better at encoding brain data than all three controls. Significant improvement in brain encoding using imitation models was found in about 40% of parcels (average across subjects) when compared to the Untrained model, and in about 85% of parcels (average across subjects) when compared with the Pixel-space and Action-space control models. Brain maps showing which regions were better encoded by the imitation models than the controls can be found in Appendix 2(Figure 8).

These results showed that neither pixel input nor the actions of the player were sufficient to capture brain dynamics in a complex task like playing Shinobi. We also observed that training an ANN with an imitation model was beneficial to brain encoding, while features derived from an untrained CNN-LSTM network (to which raw input was fed) did not explain any more variance than models trained on raw input game pixel data. Taken together, these control experiments showed that neither the stimuli, the final behavior nor the network architecture alone might successfully capture the brain dynamics. Instead, a deep learning model that learns to perform the task in a way similar to the subject better encodes the brain dynamics of that subject.

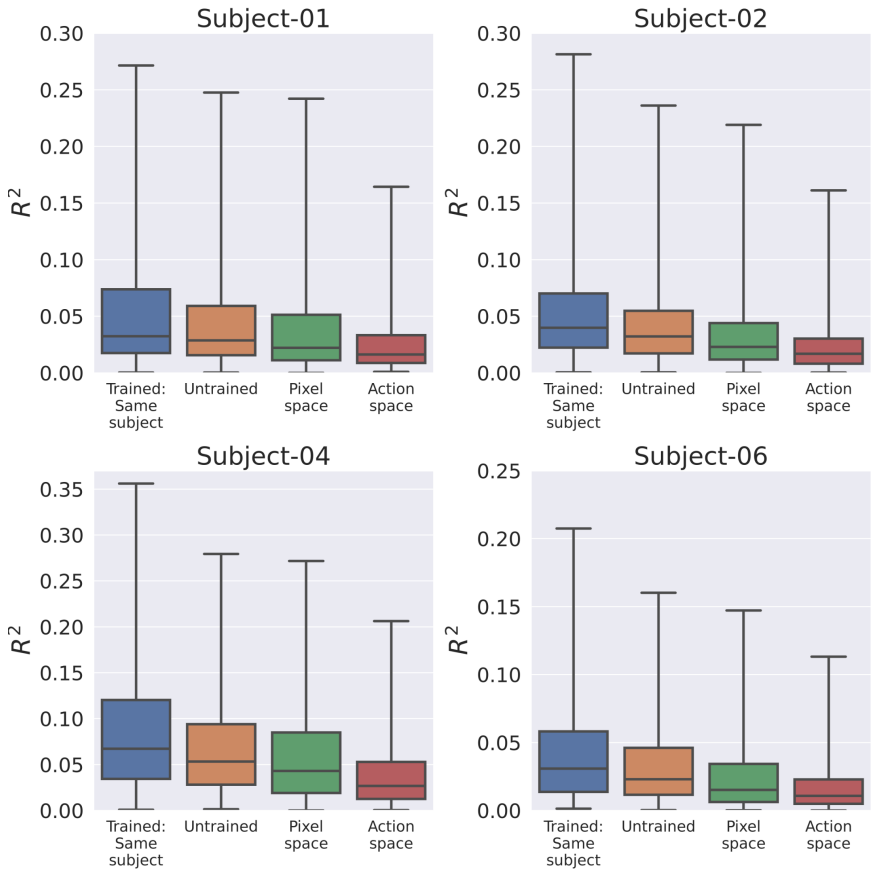


**Figure 4.** Distribution of brain $R^2$ comparing encoding performed using model trained on target(same) subject vs three control models.

## Conclusion

In this work, we showed that a deep recurrent neural network was able to successfully imitate the actions of a human player in a complex videogame and that the representations learned by this network could be used to perform brain encoding. Those individual imitation models were subject-specific, both in terms of behavior (button presses) and brain encoding. Subject-specific imitation models also performed significantly better than a series of control models. Specifically, we found that the improvement in encoding was particularly marked in motor, somatosensory, and visual cortices.

One notable limitation of this study is that given the slow temporal resolution of fMRI and the quick nature of video gameplay, it is possible that the advantages of a subject-specific imitation learning model might not have been completely evident due to a loss of fast action dynamics. Future work using high temporal resolution imaging techniques, such as EEG or MEG, would thus complement this fMRI study. The current work was still required to establish which brain regions can be successfully encoded with videogame play, an information that fMRI is better suited to provide than M/EEG.

This work highlights the potential of combining imitation learning, brain imaging, and videogames to uncover subject-specific relationships between brain and behavior.

## Materials and Methods

The summary of our experiment pipeline is shown in Figure 1. In the sections below, we describe in detail, the dataset and its collection (STEP-1 in Figure 1), followed by a description of the Imitation learning neural network and its training (STEP-2) and then how the brain encoding was performed (STEP-3).

### Dataset

The shinobi dataset used in this project was obtained from the Courtois-Neuromod databank ***Boyle et al.*** (***2020***). The shinobi dataset consists of both fMRI and behavioral data acquired while participants played the videogame Shinobi III: Return of the Ninja Master (Sega, 1993). Data was collected on four right-handed subjects (2 women) aged from 33 to 49 years old during acquisition.

#### Shinobi videogame

In this task, the subjects had to play Shinobi III: Return of the Ninja Master, an action-scrolling platformer in which the player incarnates a ninja trying to free their village from a fictitious group of mercenaries. We selected three levels of the game for their gameplay similarity; levels 1, 4, and 5. In these levels, a player starts on the far left of the map and has to reach and complete a boss-fight located at the far right of the map. To do so, a player can move, crouch, jump, and hit enemies while collecting bonuses and avoiding various obstacles. The full completion of a level usually takes about 2 to 3 minutes, and a repetition of the level was considered complete when a) a player successfully beat the boss of that level, or b) a player dies.

The game was emulated using OpenAI's gym-retro ***Nichol et al.*** (***2018***), a toolbox that allowed the extraction of game frames and memory state variables, and presented using PsychoPy ***Peirce et al.*** (***2019***).

#### Experiment setup

In phase one of the experiment participants were first instructed to play outside of the MRI scanner our version of the Shinobi video game using a computer monitor or TV. Subjects played until they reached a proficient level of skill, determined using performance metrics such as maximum score reached and speed of level completion.

Once subjects reached the proficient level of play they were invited to our MRI facility to play more sessions of the shinobi task for phase two. We collected behavioral data (videogame frames and controller key presses) and fMRI BOLD activation data as they played in the scanner. Around 10 sessions per subject were recorded with each session lasting for about 40 minutes and composed of multiple runs. One run of the Shinobi task comprises sequential repetitions of levels 1, 4, and 5 (in this order) looping back to level 1 when the first three levels are completed. The runs lasted for at least 10 minutes and ended when the current repetition ended after the 10 minutes were elapsed.

For this project, only Phase two data (behavioral and fMRI) was used. Few runs and sessions were not usable based on quality checks and hence were excluded.

The data used per subject is shown in Table 1.

| Subject | Total sessions | Total runs | Data (in hours) |
|---|---|---|---|
| 01 | 12 | 53 | 7.20 |
| 02 | 8 | 37 | 5.41 |
| 04 | 8 | 40 | 6.02 |
| 06 | 10 | 43 | 6.35 |

**Table 1.** dataset details

### fMRI data

MRI data was collected using a 3T Siemens Prisma Fit MR scanner, a 64-channel head coil. fMRI data was collected using an accelerated simultaneous multi-slice imaging sequence (***Setsompop et al., 2012***) with a spatial resolution of 2mm isotropic, and a Repetition Time (TR) of 1.49s. Visual stimuli were projected onto a screen via a waveguide and presented to participants on a mirror mounted on the head coil. For a more detailed description of the MRI or fMRI sequences and set up are described at Courtois Neuromod's Project documentation page: https://docs.cneuromod.ca.

### Data Preprocessing

*Behavioral data* was recorded at 60Hz. However, it was observed that humans play the game at about 10 to 15Hz (min press duration of 4-6 samples), which translated to each controller key press being repeated 4 to 6 times in our recordings. Hence, we downsampled the behavioral data to 12Hz. This resulted in having 18 videogame frames + controller key presses for every 1.49 seconds (Duration of each fMRI TR ). Videogame frames were converted from RGB to grayscale images. Further, we also cropped out the top portion of the videogame frame where the game score is displayed.

*fMRI data* was preprocessed using the fMRIprep pipeline(***Esteban et al., 2019***), using the "long term support" version 20.2.3. A high pass filter of 0.01 Hertz and Spatial smoothing of fwhm = 8mm was applied to the data. Further was further cleaned by using [motion + wm_csf + global] confounds using load_confounds package. Finally, voxel space fMRI data was converted into parcel space using MIST- 444 atlas (***Urchs et al., 2019***).

## Imitation learning

One approach is to follow what enables humans to learn, to learn by seeing others. So a solution could be to get an expert to do the desired behavior and make our policy imitate the expert. This would fall under the category of Reinforcement learning algorithms known as Imitation learning (IL). In this study, we want to see if training a network to play with the gameplay style of the subject better encodes the subject's brain data, and we use imitation learning to train our behavior encoding model.

### Model architecture

We use a Deep neural network-based imitation learning model. The network architecture of our model is shown in Appendix Figure 5. As the task here is to take in a videogame frame and learn which action to take such that it imitated the subject's actions, the model needs to process images as well as keep track of its actions taken at the previous timestep. Hence our imitation learning model is composed of four stacked Convolutional layers, followed by four long short-term memory (LSTM) layers with outputs from the LSTM passed into a fully connected (FC) layer that computes softmax probabilities over the action space. There are $2^7 = 128$ possible actions from all possible combinations of 7 button presses.

The architecture of our imitation model is shown in Figure 5 and layer details are described in Table 2.

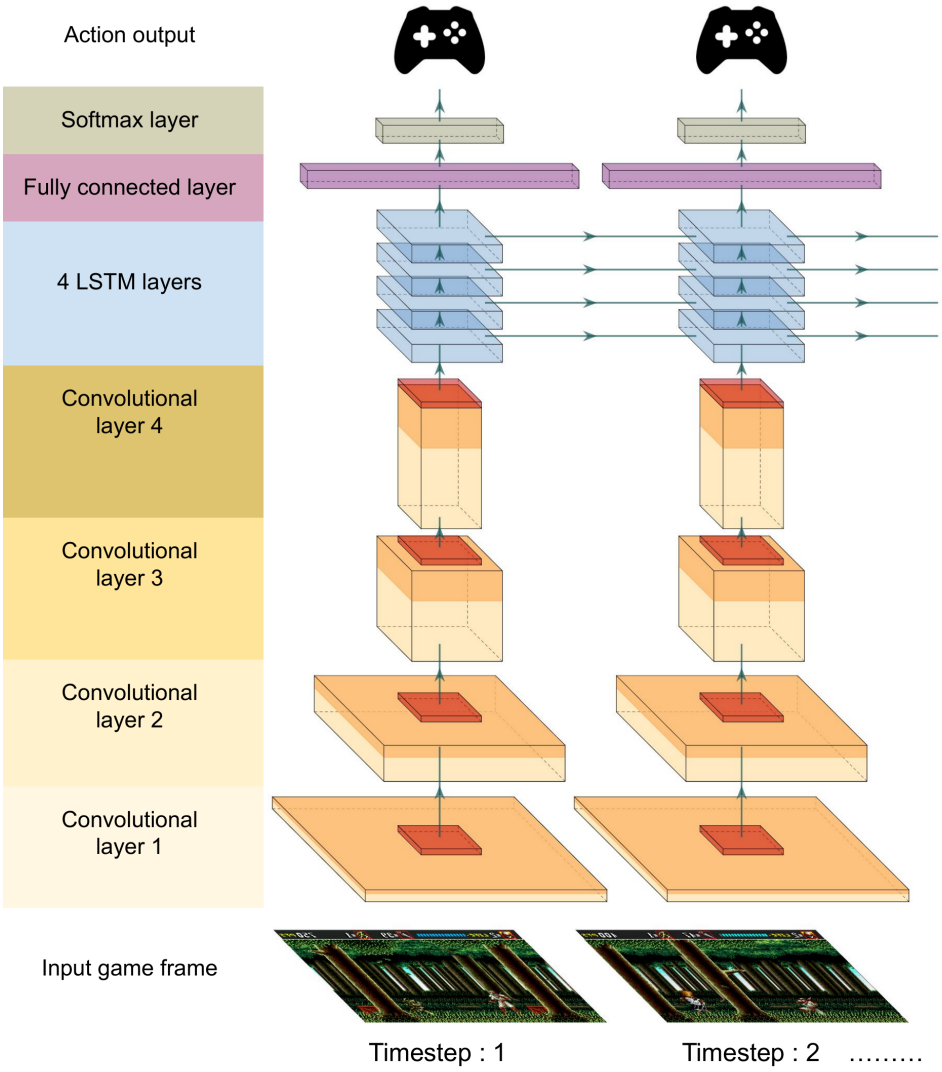


**Figure 5.** Model Architecture composed of four Convolutional layers, four LSTM layers, fully connected layer to output softmax probabilities over the action space.

| Layer | Type | Filter Size | Number of filters | Output Size |
|---|---|---|---|---|
| 0 | Input | - | - | [1,50,100] |
| 1 | Conv+Relu | [5,5] | 32 | [32,16,32] |
| 2 | Conv+Relu | [5,5] | 64 | [64,12,14] |
| 3 | Conv+Relu | [5,5] | 256 | [256,4,5] |
| 4 | Conv+Relu | [4,5] | 200 | [200,1,1] |
| 8 | Flatten | - | - | [200] |
| 9 | 4 LSTM Layers | - | - | [200] |
| 10 | Fully connected layer | - | - | [64] |
| 11 | Softmax layer | - | - | [1] |

**Table 2.** Architecture details

### Training

*Behaviour Cloning*: The simplest and one of the earliest approaches to IL is Behaviour Cloning (BC). BC aims to learn the expert's policy by treating it as a supervised learning problem. So given the expert state-action pairs, we treat them as i.i.d examples and apply supervised learning by minimizing the loss between the model's predicted action and the subject's recorded action for that game frame.

*Loss function*: There is a prominent class imbalance in the action sequences, notably the "go-right" action is the most frequent step (due to the nature of a right-scroller game). This class imbalance cannot be solved by taking subject of each actions, as a gameplay trajectory requires every action in a particular sequence, and removing some actions would result in an entirely different trajectory. Hence we used a weighted negative log-likelihood loss function (based on the frequency of each action in the data) so that errors on frequent actions are less penalized than mistakes on infrequent actions. This allows the model to focus more on the infrequent actions while maintaining the original game trajectory.

*Implementation*: The model was implemented using PyTorch 1.6 in Python 3.6. The loss was then backpropagated through the model and weights were updated using the Adam optimizer ***Kingma and Ba*** (***2019***). Early stopping ***Zhang et al.*** (***2005***) was employed to prevent overfitting and the best model was stored for validation. Hyperparameter selection is important to train a performant deep-learning model. We carried out an exhaustive search for the optimal hyperparameters following well-accepted principles and approaches in the field ***Smith*** (***2018***). The hyperparameter values found are provided in Table 3.

| Hyperparameter | Value |
|---|---|
| LSTM sequence length | 18 |
| Batch size | 50 |
| Training epochs | 3000 |
| Learning rate | 0.00005 |

**Table 3.** Training Hyperparameters

## Brain encoding

To analyze the relationship between the feature space learned by the ANN and the subject's brain data (***Wang et al., 2015***; ***Mohsenzadeh et al., 2019***). we performed brain encoding approach using a linear mapping.

The details of this analysis are summarised in Figure 6. First, we reduced the dimensions of the activations as we have limited data points and hence cannot use very large feature vector dimensions to train the linear regression model. Activations from the four CNN layers, LSTM layer, and FC layer are reduced in dimension (output vector dimension - 800) using a principal component analysis (PCA). Second, in order to get encoding features at the fMRI sampling rate, we applied another PCA (output vector dimension - 300) to compress information from 1TR=1.49-second windows, independently for each layer. Finally, the PCA vectors resulting from the second PCA from each layer are concatenated to compose the final feature space vector used for ridge regression. To account for the hemodynamic delay, the final feature space vector of the imitation model at time t is mapped to fMRI data at time t+X seconds, where the best encoding was found to be at X=6. A separate ridge regression model is fit for each parcel.

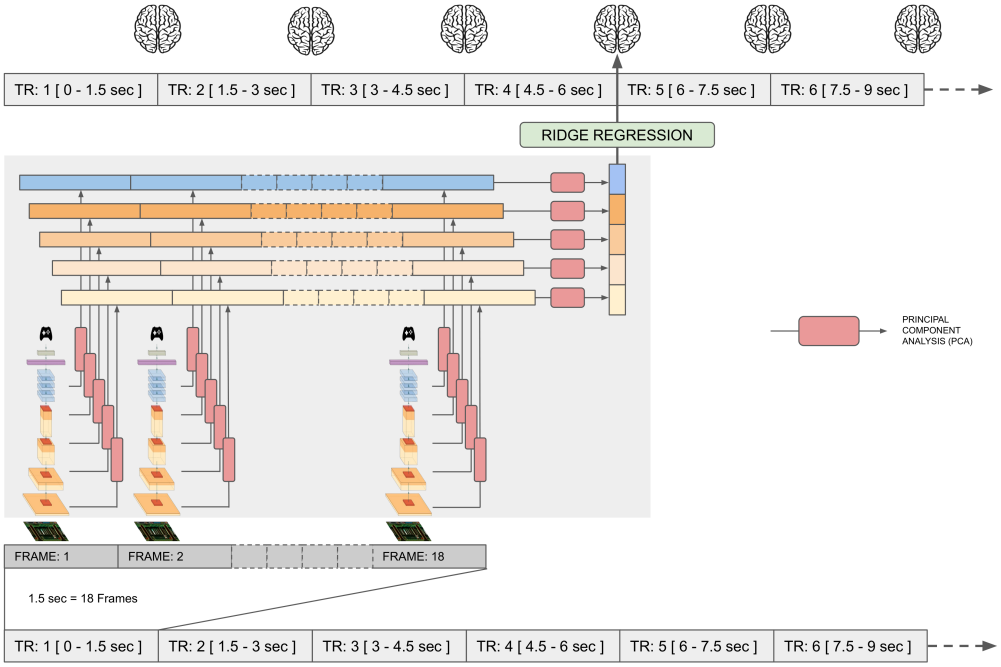


**Figure 6.** Brain encoding framework. Subject's gameplay frames are passed into a trained imitation learning model and its internal network activations are reduced in dimension using PCA. These resulting features are used to predict fMRI BOLD responses using ridge regression.

## Control models

To test whether training an ANN to mimic the behavior of a given individual endows it with higher brain encoding performance in that individual, we compared brain encoding performance across a range of control models. The models we used for comparison are described below.

*Untrained model:* When the pixel input is passed through an untrained model, due to the nature of Convolutional and LSTM layers, useful features can be captured. To understand how much of variance is captured just due to the architecture of the model, we use an untrained model with randomly initialized weights as one of our control models.

*Pixel space PCA model:* To capture the amount of brain data variance explained just by the videogame frames, we applied a PCA on the pixel RGB input (output vector dimension - 1000) and use it to train the ridge regression model.

*Action space model* As our final control model, we use the controller key presses of the subject to predict the brain data. This allows us to capture the amount of brain data variance explained just by the final action space.

## Acknowledgments

The Courtois project on neural modelling was made possible by a generous donation from the Courtois foundation, administered by the Fondation Institut Gériatrie Montréal at CIUSSS du Centre-Sud-de-l'île-de-Montréal and University of Montreal. The Courtois NeuroMod team is based at "Centre de Recherche de l'Institut Universitaire de Gériatrie de Montréal", with several other institutions involved. See the cneuromod documentation for an up-to-date list of contributors (https://docs.cneuromod.ca).

AK was supported in part by a Mitacs bursary in addition to Courtois foundation grant, FP, BP, PS, MLC, and JB were entirely supported by Courtois foundation grant, PB is supported by a salary award of the "fonds de recherche du Québec - Santé" (chercheur boursier senior). KJ is supported by funding from the Canada Research Chairs program and a Discovery Grant from the Natural Sciences and Engineering Research Council of Canada.

## References

**Anderson JR**, Bothell D, Fincham JM, Anderson AR, Poole B, Qin Y. Brain regions engaged by part-and whole-task performance in a video game: a model-based test of the decomposition hypothesis. Journal of cognitive neuroscience. 2011; 23(12):3983–3997.

**Bavelier D**, Achtman RL, Mani M, Föcker J. Neural bases of selective attention in action video game players. Vision research. 2012; 61:132–143.

**Bellec P**, Boyle J. Bridging the gap between perception and action: the case for neuroimaging, AI and video games. . 2019; .

**Boyle J**, Pinsard B, Boukhdhir A, Belleville S, Brambatti S, Chen J, Cohen-Adad J, Cyr A, Fuente, Rainville P, Bellec P. The Courtois project on neuronal modelling – first data release. 26th Annual Meeting of the Organization for Human Brain Mapping, 2020. . 2020; .

**Caucheteux C**, King JR. Brains and algorithms partially converge in natural language processing. Communications biology. 2022; 5(1):1–10.

**Cross L**, Cockburn J, Yue Y, O'Doherty JP. Using deep reinforcement learning to reveal how the brain encodes abstract state-space representations in high-dimensional environments. Neuron. 2020; http://www.sciencedirect.com/science/article/pii/S0896627320308990, doi: https://doi.org/10.1016/j.neuron.2020.11.021.

**Cross L**, Cockburn J, Yue Y, O'Doherty JP. Using deep reinforcement learning to reveal how the brain encodes abstract state-space representations in high-dimensional environments. Neuron. 2021; 109(4):724–738.

**DiCarlo JJ**, Zoccolan D, Rust NC. How does the brain solve visual object recognition? Neuron. 2012; 73(3):415–434.

**Esteban O**, Markiewicz CJ, Blair RW, Moodie CA, Isik AI, Erramuzpe A, Kent JD, Goncalves M, DuPre E, Snyder M, et al. fMRIPrep: a robust preprocessing pipeline for functional MRI. Nature methods. 2019; 16(1):111–116.

**Kell AJE**, Yamins DLK, Shook EN, Norman-Haignere SV, McDermott JH. A Task-Optimized Neural Network Replicates Human Auditory Behavior, Predicts Brain Responses, and Reveals a Cortical Processing Hierarchy. Neuron. 2018 May; 98(3):630–644.e16. https://doi.org/10.1016/j.neuron.2018.03.044, doi: 10.1016/j.neuron.2018.03.044.

**Kingma DP**, Ba JA. A method for stochastic optimization. arXiv 2014. arXiv preprint arXiv:14126980. 2019; 434.

**Mohsenzadeh Y**, Mullin C, Pantazis D, Oliva A, Emergence of Topographical Correspondences between Deep Neural Network and Human Ventral Visual Cortex; 2019.

**Nichol A**, Pfau V, Hesse C, Klimov O, Schulman J. Gotta Learn Fast: A New Benchmark for Generalization in RL. arXiv preprint arXiv:180403720. 2018; .

**Peirce J**, Gray JR, Simpson S, MacAskill M, Höchenberger R, Sogo H, Kastman E, Lindeløv JK. PsychoPy2: Experiments in behavior made easy. Behavior research methods. 2019; 51(1):195–203.

**Schrimpf M**, Kubilius J, Hong H, Majaj NJ, Rajalingham R, Issa EB, Kar K, Bashivan P, Prescott-Roy J, Geiger F, et al. Brain-score: Which artificial neural network for object recognition is most brain-like? BioRxiv. 2018; p. 407007.

**Setsompop K**, Cohen-Adad J, Gagoski BA, Raij T, Yendiki A, Keil B, Wedeen VJ, Wald LL. Improving diffusion MRI using simultaneous multi-slice echo planar imaging. Neuroimage. 2012; 63(1):569–580.

**Smith LN**. A disciplined approach to neural network hyper-parameters: Part 1 - learning rate, batch size, momentum, and weight decay. CoRR. 2018; abs/1803.09820. http://arxiv.org/abs/1803.09820.

**Srirangarajan T**, Mortazavi L, Bortolini T, Moll J, Knutson B. Multi-band FMRI compromises detection of mesolimbic reward responses. NeuroImage. 2021; 244:118617.

**Urchs S**, Armoza J, Moreau C, Benhajali Y, St-Aubin J, Orban P, Bellec P. MIST: A multi-resolution parcellation of functional brain networks. MNI Open Research. 2019; 1:3.

**Wang P**, Malave V, Cipollini B. Encoding voxels with deep learning. Journal of Neuroscience. 2015; 35(48):15769–15771.

**Zador A**, Richards B, Ölveczky B, Escola S, Bengio Y, Boahen K, Botvinick M, Chklovskii D, Churchland A, Clopath C, et al. Toward next-generation artificial intelligence: catalyzing the NeuroAI revolution. arXiv preprint arXiv:221008340. 2022; .

**Zhang T**, Yu B, et al. Boosting with early stopping: Convergence and consistency. The Annals of Statistics. 2005; 33(4):1538–1579.

## Appendix 1 - Layer wise analysis

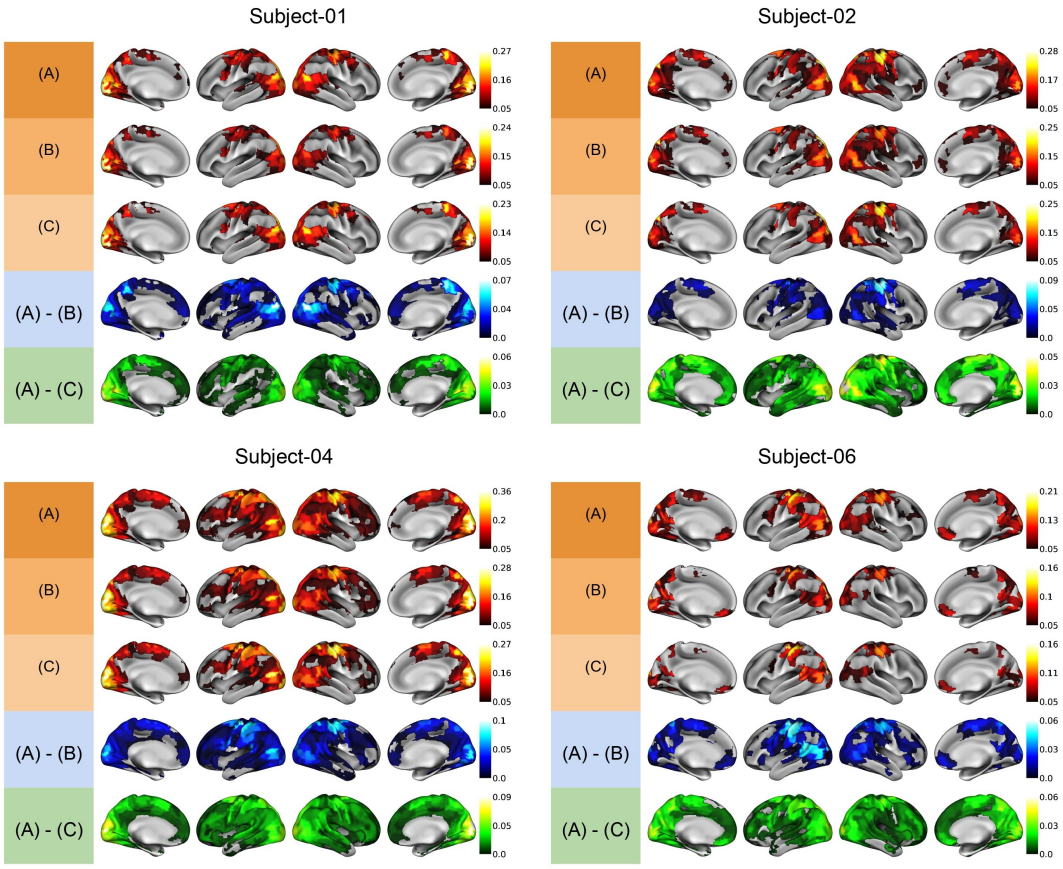


**Figure 7.** $R^2$ values on a brain map shown for each of the four subjects. Comparison of encoding based on which layers from the trained model were used. **A**: All the layers features were used. **B**: Only early two convolutional layer features were used. **C**: Last convolutional layer and LSTM layer features were used. **A-B**: $R^2$(All layers) - $R^2$(Early layers). This shows the effect of the latter layers in the model. **A-C**: $R^2$(All layers) - $R^2$(Last layers). This shows the effect of early layers in the model.

In this section, we analyze what effect the early layers vs later layers of the model have on brain encoding and which brain regions each of them help to encode better. Apart from the encoding model that uses all the layer features we consider two additional models. Early layer model, where only the initial two layers (Convolutional layer-1, Convolutional layer-2) features were used, and last layers model where only the last two layers (Convolutional layer-4 and LSTM layer) features were used to predict the brain data using ridge regression.

In order to see which regions of the brain were predicted better by which layers of the model, we projected the $R^2$ values in the brain map for each subject as shown in Figure 7. We found that the early layer predicted the early visual cortex activation better (shown in Figure 7 by rows (B) and (A-C)), and the latter layers of the model better predicted the higher visual cortex, motor, and somatosensory regions (shown in the Figure 7 by rows (C) and (A-B)). This can be attributed to the fact that early convolutional layers capture the primitive visual features of the image similar to what happens in the early visual cortex areas, and the latter layers of the model capture the gameplay behavior, which is also exhibited in motor and somatosensory regions of the brain. We see that across all four subjects, encoding performed using all the layers features was better at predicting the brain data rather than using just the early layer features or just the latter layer features.

## Appendix 2 - Control analysis brain maps

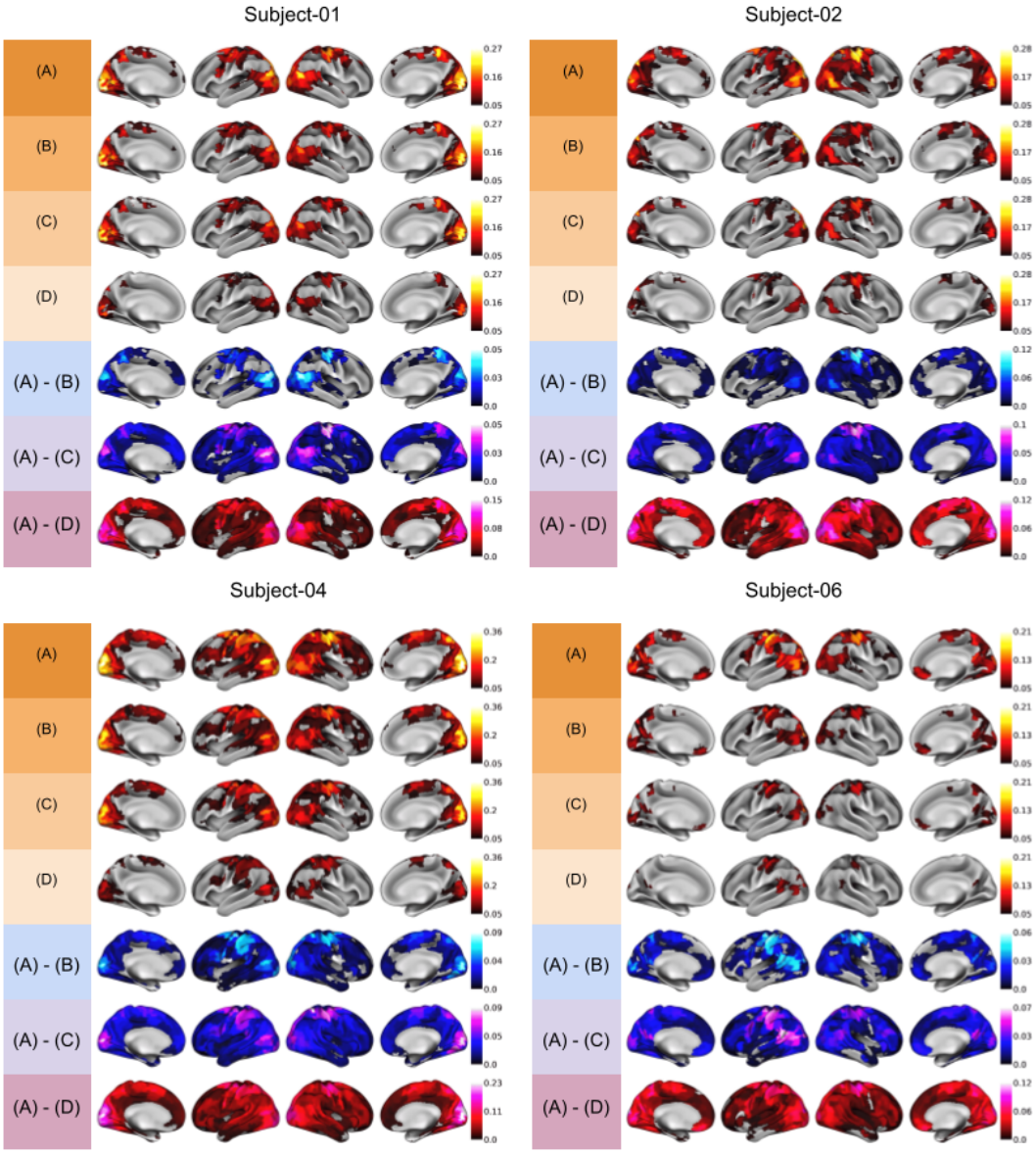


**Figure 8.** $R^2$ values on a brain map shown for each of the four subjects. **A**: Trained model (Same subject gameplay). **B**: Untrained model. **C**: Pixel-space model. **D**: Action-space model. **A-B**: $R^2$(Trained model) - $R^2$(Untrained model). **A-C**: $R^2$(Trained model) - $R^2$(Pixel space model). **A-D**: $R^2$(Trained model) - $R^2$(Action space model).