nature machine intelligence

Article

https://doi.org/10.1038/s42256-024-00802-0

Learning high-level visual representations from a child's perspective without strong inductive biases

Received: 24 May 2023

Accepted: 5 February 2024

Published online: 7 March 2024

Check for updates

A. Emin Orhan [1] ✉ & Brenden M. Lake [1,2]

Young children develop sophisticated internal models of the world based on their visual experience. Can such models be learned from a child's visual experience without strong inductive biases? To investigate this, we train state-of-the-art neural networks on a realistic proxy of a child's visual experience without any explicit supervision or domain-specific inductive biases. Specifically, we train both embedding models and generative models on 200 hours of headcam video from a single child collected over two years and comprehensively evaluate their performance in downstream tasks using various reference models as yardsticks. On average, the best embedding models perform at a respectable 70% of a high-performance ImageNet-trained model, despite substantial differences in training data. They also learn broad semantic categories and object localization capabilities without explicit supervision, but they are less object-centric than models trained on all of ImageNet. Generative models trained with the same data successfully extrapolate simple properties of partially masked objects, like their rough outline, texture, colour or orientation, but struggle with finer object details. We replicate our experiments with two other children and find remarkably consistent results. Broadly useful high-level visual representations are thus robustly learnable from a sample of a child's visual experience without strong inductive biases.

Young children develop powerful internal models of the visual world. Their visual abilities for object categorization[1,2], segmentation[3] and physical prediction[4] emerge well within the first year. By the time children are 4–5 years old, their object recognition capabilities are already mature enough that they can outperform highly capable computer vision models in challenging real-world visual object recognition tasks in head-to-head comparisons[5,6].

Is it possible to learn such powerful internal models of the world from a child's experience without strong, domain-specific inductive biases? Versions of this 'nature versus nurture' question have been debated for centuries[7,8], and they continue to shape our understanding of intelligence. In the last couple of decades, some developmental psychologists hypothesized various innate inductive biases related to objects, agents and space[3,4,9], as well as biases governing the categorization and labelling of objects[10,11]. Others, on the other hand, argued for the feasibility of building internal models of the world without such inductive biases, relying instead on the richness of the developing child's experience[12].

Here we approach this age-old 'nature versus nurture' question through a modern lens: we investigate what today's highly generic deep neural networks can learn from a representative sample of a child's egocentric visual experience. We train state-of-the-art self-supervised learning (SSL) algorithms on a large-scale, longitudinal, developmentally realistic dataset of headcam videos recorded from the perspective

[1]Center for Data Science, New York University, New York, NY, USA. [2]Department of Psychology, New York University, New York, NY, USA.
✉e-mail: eo41@nyu.edu

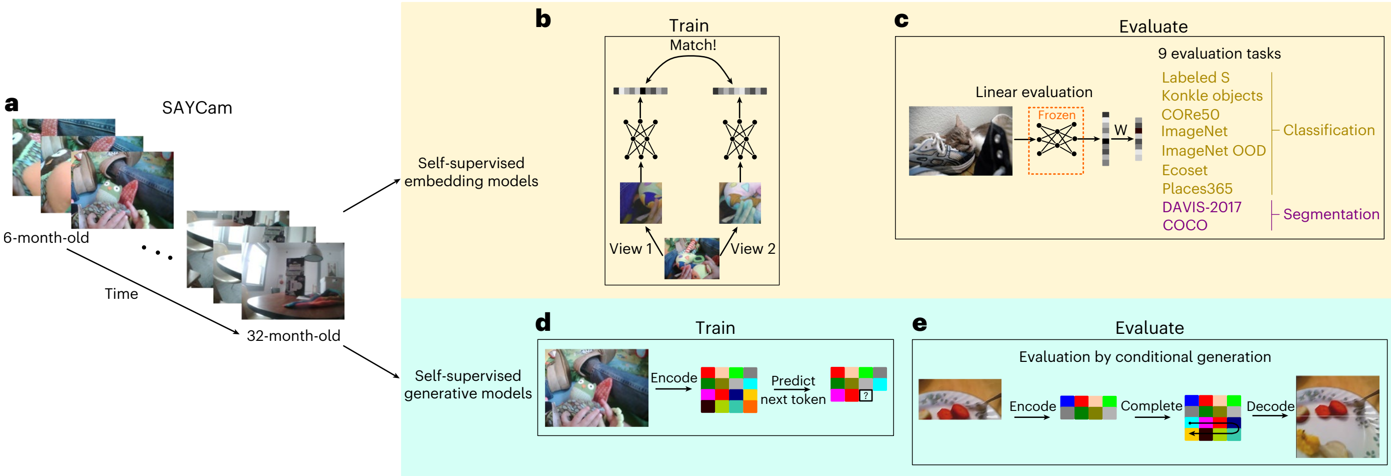

Fig. 1 | Schematic overview of the experiments. a, Example video frames from longitudinal headcam recordings from one of the children in SAYCam[13]. **b**, Training self-supervised embedding models. For purposes of illustration, only a self-distillation type SSL algorithm is shown, where the high-level goal is to learn representations that are similar across different views of the same image. **c**, Evaluating the self-supervised embedding models. We evaluate the learned representations by training lightweight readouts on top of frozen features in nine downstream classification or segmentation tasks. **d**, Training self-supervised generative models. Frames are encoded into a spatially downsampled discrete code with the help of an optimized codebook. An autoregressive transformer model is trained to predict the next token in the discrete code. **e**, Evaluating the self-supervised generative models. The top half of an evaluation image is given as context to the model. The model completes the bottom half of the image in the latent space, and the model-completed latent code is decoded back to the image space for evaluation.

of individual children[13]. The dataset comprises hundreds of hours of longitudinal, natural videos recorded over 26 months of early development. Distinctive to our work, we train models on data from each individual child, simulating the child's learning problem as closely as possible. By using highly generic architectures and learning algorithms, we seek to understand what kinds of perceptual capabilities might be learnable from a child's visual experience without strong inductive biases.

We train image embedding models that can be used in a variety of downstream visual recognition, segmentation, or detection tasks, and generative models that can be used to generate images and assign likelihoods to them. We quantitatively evaluate the capabilities of the trained models, compare their performance against a battery of reference models and provide qualitative insights into the properties of the learned representations.

Models

We train the two distinct types of models—embedding models and generative models—on a representative sample of a child's visual experience. Embedding models aim to learn high-level visual features that are useful for a variety of downstream visual tasks. Generative models can generate novel images (both conditional on a given context and unconditionally) and assign likelihoods to images, providing a complementary tool for examining the acquired knowledge. Here we briefly describe the algorithms, architectures, training and evaluation methods relating to these models (Fig. 1). Methods provides additional details.

Embedding models

Self-supervised learning algorithms

SSL algorithms seek to learn useful, high-level representations from a dataset without using any explicit supervision signals like semantic labels. Instead, they use augmented views of the training examples to generate self-supervision signals (Fig. 1b). We train embedding models with three different visual SSL algorithms: DINO[14], Mugs[15] and masked autoencoders (MAEs)[16].

Model architectures

Since our goal is to address a question of learnability with minimal inductive biases, we choose highly generic model architectures with minimal inductive biases. In particular, we focus mainly on vision transformer (ViT) models[17]. We train models in three standard sizes: ViT-S, ViT-B, ViT-L (with approximately 21 million, 85 million, 306 million parameters, respectively), all with 16 × 16 patches. With DINO, we further train ViT-B models with 14 × 14 patches, as well as a convolutional ResNeXt-50 (32x4d) model[18] with 25 million parameters.

The ViT models and the ResNeXt model incorporate two main inductive biases: hierarchical composition and translation invariance. These are very generic inductive biases quite different from the stronger, more domain-specific inductive biases about language, objects, agents, categories, or places that are sometimes hypothesized by psychologists. The ResNeXt model incorporates a further spatial inductive bias with its convolutional filters. Our implementation of the ViT models, on the other hand, uses learned position embeddings that are initialized randomly, therefore the ViT models effectively start out with no spatial inductive biases.

Training data

Our main goal is to evaluate what can be learned from a sample of the visual experience of a developing child. To this end, we use the SAYCam dataset[13], a large-scale, longitudinal dataset of natural headcam videos recorded from the perspective of three young children (S, A and Y) between the ages of 6 to 31 months (Fig. 1a). The dataset contains 194 hours of video from S (6–30 months), 141 hours of video from A (8–31 months) and 137 hours of video from Y (7–24 months) for a total of 472 hours of video. Data from each child consist of a series of continuous headcam recordings, usually 1–2 hours of recording per week. These contain both indoor and outdoor recording episodes. Videos are subsampled at five frames per second, yielding 9 million frames across three children. We train models on data from each child individually as well as on the combined data (denoted as SAY below). Further details regarding the dataset can be found in ref. 13.

Reference models

To compare SAYCam-learned representations with representations learned from static photographic images, we train ViT-B/14 models (with DINO) on ImageNet[19] and randomly sampled subsets of ImageNet (100%, 10% and 1% of the training set). To compare SAYCam-learned

https://doi.org/10.1038/s42256-024-00802-0

representations with representations learned from other video datasets, we train ViT-B/14 models (with DINO) on 200-hour-long subsets of Kinetics-700 (ref. 20) and Ego4D[21] datasets (denoted as Kinetics-200h and Ego4D-200h below). Kinetics-700 consists of very short YouTube clips of people performing various actions, whereas Ego4D consists of long, continuous, egocentric headcam recordings from adults. We finally consider a randomly initialized, untrained reference model with the same architecture as the other reference models (ViT-B/14).

Evaluation

We use seven different classification tasks and two different semantic segmentation tasks for evaluation (see Fig. 1c for the full list). These include a classification task based on a labelled subset of the data from child S in SAYCam (Labeled S), common object recognition (ImageNet) and image segmentation (COCO) benchmarks as well as a place classification task (Places365). Using a wide range of evaluation tasks and datasets allows us to arrive at a more complete and robust picture of the overall quality of the learned visual representations. To evaluate visual representations learned exclusively through SSL, we use either completely non-parametric evaluation methods or methods that involve learning only a single layer of learnable parameters on top of frozen features (Fig. 1c).

Generative models

Self-supervised learning algorithm

We train generative autoregressive transformer models on child headcam data. We first learn a discrete codebook with a vector quantized generative adversarial network (VQGAN)[22] and then encode each video frame as a spatial grid of integers from the codebook. These codes are then flattened and fed into a generative pretrained transformer (GPT) model to learn a prior over the video frames. The GPT model is trained with the standard autoregressive language modelling objective[23], that is, predicting the next token given all previous tokens in the flattened code (Fig. 1d). We refer to the entire combined model as a VQGAN-GPT model.

Evaluation

We consider conditional generation tasks where we take evaluation images, give the upper half of each image as context and ask the model to complete the bottom half of the image conditional on the upper half (Fig. 1e).

Results

Embedding models

Quantitative summary. Figure 2 summarizes the evaluation results of the embedding models, singling out the effects of the SSL algorithm (Fig. 2a), model architecture (Fig. 2b) and pretraining data (Fig. 2c) on downstream task performance. In Fig. 2a–c, we normalize the performance on each task by the performance of a ViT-B/14 model trained with DINO on all of ImageNet, the overall best model. The DINO algorithm performs the best in our evaluations, with Mugs coming in second and MAE third. Different model architectures perform similarly, except for ViT-S/16, which performs worse than the other models. Given these results, we focus most of our subsequent analyses on ViT-B/14 models trained with DINO, which is one of our best model and algorithm combinations overall.

Figure 2c compares the performance of SAYCam-trained models against each of the reference models described above. Figure 2d further splits Fig. 2c into different evaluation tasks. On average, SAYCam-trained models perform at 65–70% of a model trained on the full ImageNet training set, and they are generally comparable to a model trained with 10% of ImageNet (means ± standard errors: SAY: 70.2% ± 8.0%, S: 69.7% ± 8.4%, A: 66.5% ± 7.1%, Y: 64.5% ± 7.2%, ImageNet-100%: 100.0% ± 0.0%, ImageNet-10%: 69.7% ± 6.0%). Thus, although SAYCam-trained models are exposed to a very different type of data (less diverse, temporally extended, noisy headcam videos) than the ImageNet-trained model, they are able to recover a substantial fraction of the ImageNet-trained model's performance.

All SAYCam-trained models substantially outperform the untrained reference model with random features (Random: 18.6% ± 5.7%). Differences across individual children in SAYCam are relatively small (for example, only 3% relative difference between the approximately length-matched A and Y). Finally, the Ego4D-200h model performs comparably to the models trained on A and Y and slightly worse than the model trained on the approximately length-matched S (Ego4D-200h: 65.6% ± 7.1%), whereas the Kinetics-200h model performs better than all SAYCam-trained models (Kinetics-200h: 74.5% ± 6.7%), although the difference is surprisingly small given the very different nature of the videos in Kinetics-200h compared with the videos in SAYCam or Ego4D (videos in Kinetics-200h are much shorter and more diverse in content).

The following qualitative analyses focus on models trained with the headcam data from child S only. The results for the other two children are qualitatively similar; they can be found in Supplementary Figs. 1–4.

Learning to localize semantic categories without location supervision. The semantic segmentation results in Fig. 2d (DAVIS-2017 and COCO) show visual representations learned from a child's headcam data are much better than random representations at localizing semantic categories in an image, given dense (pixel-level) semantic feedback. These representations can also support localizing semantic categories without any explicit location feedback, using only information from a linear classifier trained on a downstream classification task. The last-layer feature maps of the model can be linearly combined with the classifier weights for a given class, generating a class activation map (CAM)[24]. Figure 3a illustrates CAMs for four different categories from the Labeled S evaluation dataset. Qualitatively, the semantic localization obtained from CAMs is reasonably accurate in many, though not all, cases. Common failure cases include difficulties with localizing smaller objects and overbroad activation maps that extend into neighbouring objects or surfaces. This may be related to the relatively global, background-sensitive nature of the representations learned by models trained with the child headcam data, as discussed next.

Learning more global, background-sensitive representations. Visual representations learned from the child headcam data tend to be less object-centric and more sensitive to background and low-level surface features (for example, contours) compared to ImageNet-learned representations. This is illustrated in Fig. 3b, which compares the mean attention maps (averaged over all attention heads) of ViT-B/14 models trained on ImageNet and on the headcam data from child S. These observations are quantitatively supported by the performance of the models on CORe50 (Fig. 2d), which evaluates the background-invariance of the models' object representations. Models trained with small subsets of ImageNet are also less object-centric (Supplementary Fig. 8), suggesting that learning object-centric, background-invariant representations may require seeing the foreground objects against a sufficiently large and diverse set of backgrounds.

Learning broad semantic categories without any labelled examples. A rich semantic structure emerges in the embedding space of the models trained with the child headcam data. Figure 4 shows a t-distributed stochastic neighbor embedding (t-SNE) visualization[25] of the mean embeddings of the 1,000 ImageNet classes (estimated over the validation set) obtained from a model trained on child S. Classes belonging to the same broad semantic categories, such as dogs, birds, reptiles, insects, vehicles, musical instruments, food, clothing, and so on, tend to be clustered together in the embedding space. Notably, the model learns this structure automatically without any labelled examples. This structure is either absent or much weaker in the embedding space of untrained, random models (Supplementary Fig. 6; also see Supplementary Figs. 3–5 for embeddings from other

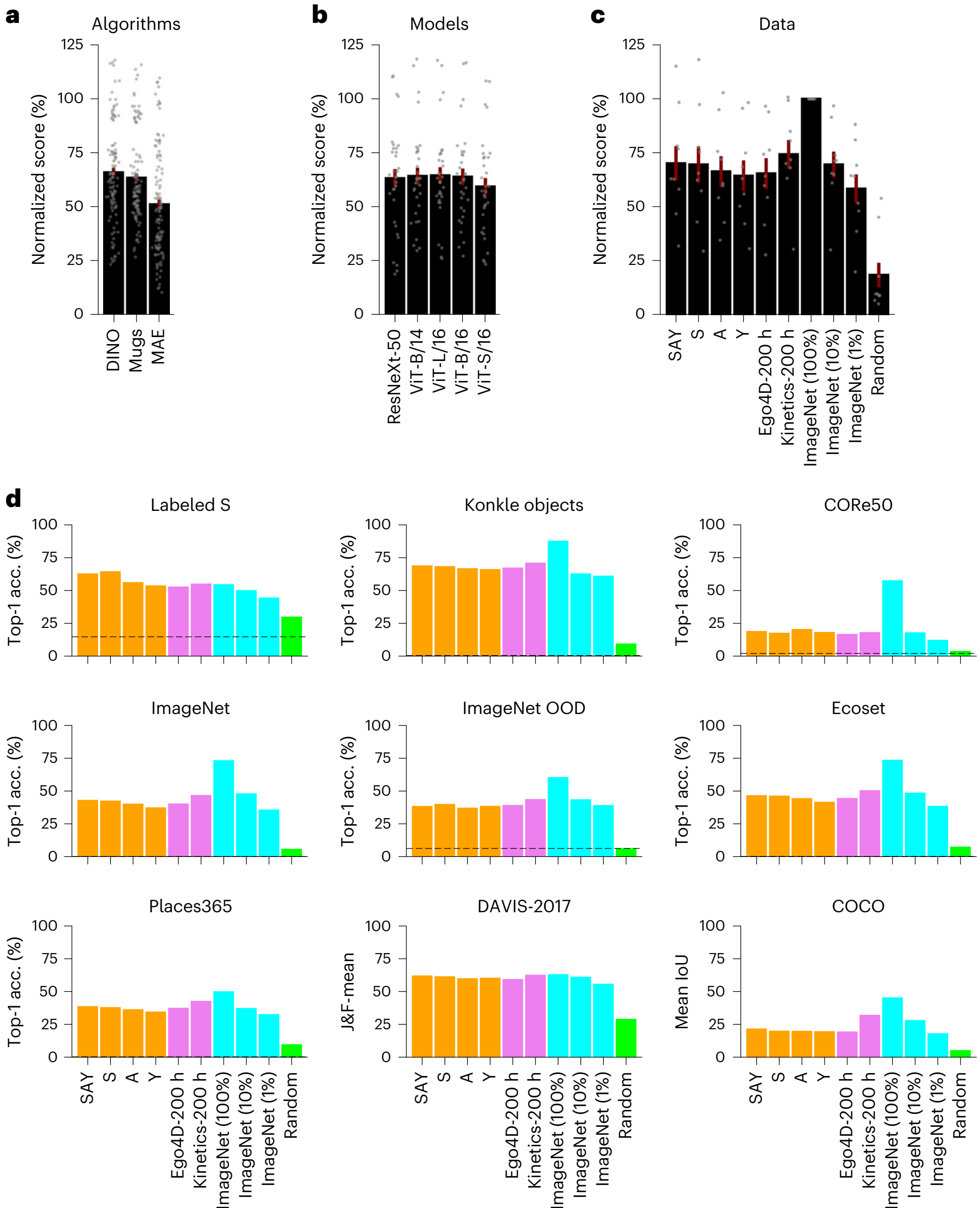

Fig. 2 | Quantitative evaluation of the embedding models. a,**b**,**c**, The effect of algorithm (**a**), model architecture (**b**) and pretraining data (**c**) on the performance in downstream evaluation tasks. All scores in **a**, **b** and **c** are relative to the ViT-B/14 model trained with DINO on all of ImageNet, our best model overall. Error bars represent standard errors. In **a**, means and standard errors are calculated over $n = 3 \times 4 \times 9 = 108$ different combinations (3 models, ViT-S/16, ViT-B/16 and ViT-L/16; 4 datasets, SAY, S, A and Y; and 9 evaluation tasks), represented by the individual grey dots. In **b**, the algorithm is fixed to DINO and the means and standard errors are calculated over $n = 8 \times 4 = 32$ different combinations (8 evaluation tasks, omitting DAVIS-2017; and 4 datasets). In **c**, the algorithm is fixed to DINO, the model architecture is fixed to ViT-B/14, and the means and standard errors are calculated over $n = 9$ evaluation tasks. **d**, Performance of SAYCam-trained models compared with the reference models in all 9 evaluation tasks. As in **c**, here we again fix the algorithm to DINO and the model architecture to ViT-B/14. SAYCam-trained models are shown in orange; models trained on other video datasets are shown in magenta; ImageNet-trained models are shown in cyan; and the untrained reference model is shown in green. Dashed horizontal lines show chance-level performance for the classification tasks. Note that performance is not normalized in **d**. acc., accuracy; J&F, region and contour similarity; IoU, intersection over union.

trained models). Interestingly, the semantic structure that emerges in the embedding spaces of SAYCam-trained models is representationally most similar to the semantic structure in a model trained with the egocentric headcam data from adults (Ego4D-200h), followed by the other models that perform similarly in the downstream evaluation tasks (Supplementary Fig. 7).

Nearest neighbours reveal semantic structure in the embedding space. Figure 5 shows query images from the Open Images V7 dataset[26] (leftmost column) and their ten nearest neighbours in two different embedding spaces. Retrievals from the embedding space of a model trained with the headcam data from child S are often semantically related to the query image (Fig. 5a). The failure cases usually preserve some semantic relationships (for example, retrieval of horses, dogs, or other animals for the bird query in the sixth row of Fig. 5a) or display visual similarities with the texture or the overall shape of the object depicted in the query image (for example, the food item queried in the second row of Fig. 5a and the other food items retrieved in response to it have similar visual textures and/or shapes). The retrievals from the embedding space of an untrained, random model, on the other hand,

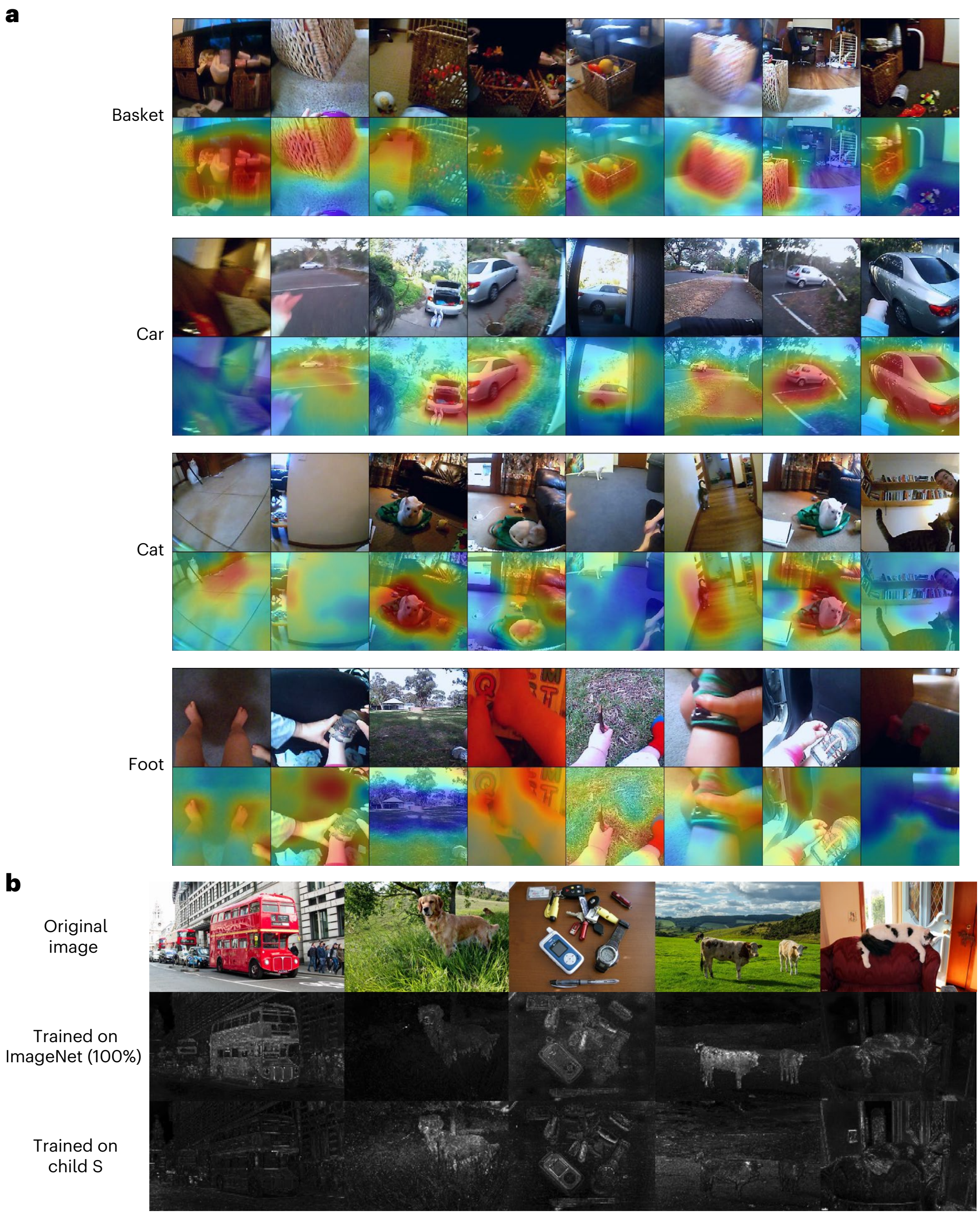

Fig. 3 | Qualitative evaluation of the embedding models. a, CAMs for four different classes in Labeled S: basket, car, cat, foot. In each case, the top row shows the original images, and the bottom row shows the corresponding class activation maps. The class activation maps shown here are from a ResNeXt-50 model trained with DINO on data from child S only. More examples can be found at the accompanying code repository. **b**, Example images and the corresponding attention maps (averaged over all attention heads) for ViT-B/14 models trained on all of ImageNet training set and on data from child S in SAYCam, respectively. The attention maps were computed with respect to the cls token. Images from Flickr. Credits (left to right): Henry Zbyszynski, Franco Vannini, John Hritz, sonder3, Lisa Zins.

seem to be primarily driven by the overall colour similarity between the query and the retrieved item (Fig. 5b).

Generative models

Generative models offer an alternative and intuitive route to studying learnability from a child's visual experience, as their outputs can be visualized directly. Here we use an image completion task to probe the visual knowledge acquired by generative models trained on the child headcam data. We provide the model with the upper half of an image and generate the bottom half from the model with sampling. Figure 6a shows different images (columns) from child Y's data together with completions generated by a model trained on another child (child S) as well as a model trained on all of ImageNet. Similarly, Fig. 6b shows different images from the Konkle objects dataset and the corresponding completions. All of these completions are 'zero-shot' in that the models have not seen any examples from these datasets during training. Although the model trained on child S can usually generate completions that match the colour, texture, orientation and rough outline of the object (or objects) given in the context (for example, the compass in Fig. 6b; second image from the right), it is not very successful at generating finer details of the objects (for example, it is not very good at generating plausible looking legs for the dog in Fig. 6b). The model trained on all of ImageNet, on the other hand, is much better at generating finer object details. We measure the quality of the completions generated by different models through Fréchet Inception Distance (FID) scores evaluated on two datasets under different conditions (see Methods and Supplementary Table 1). The FID scores broadly confirm our qualitative observations. In particular, the model trained on all

 https://doi.org/10.1038/s42256-024-00802-0

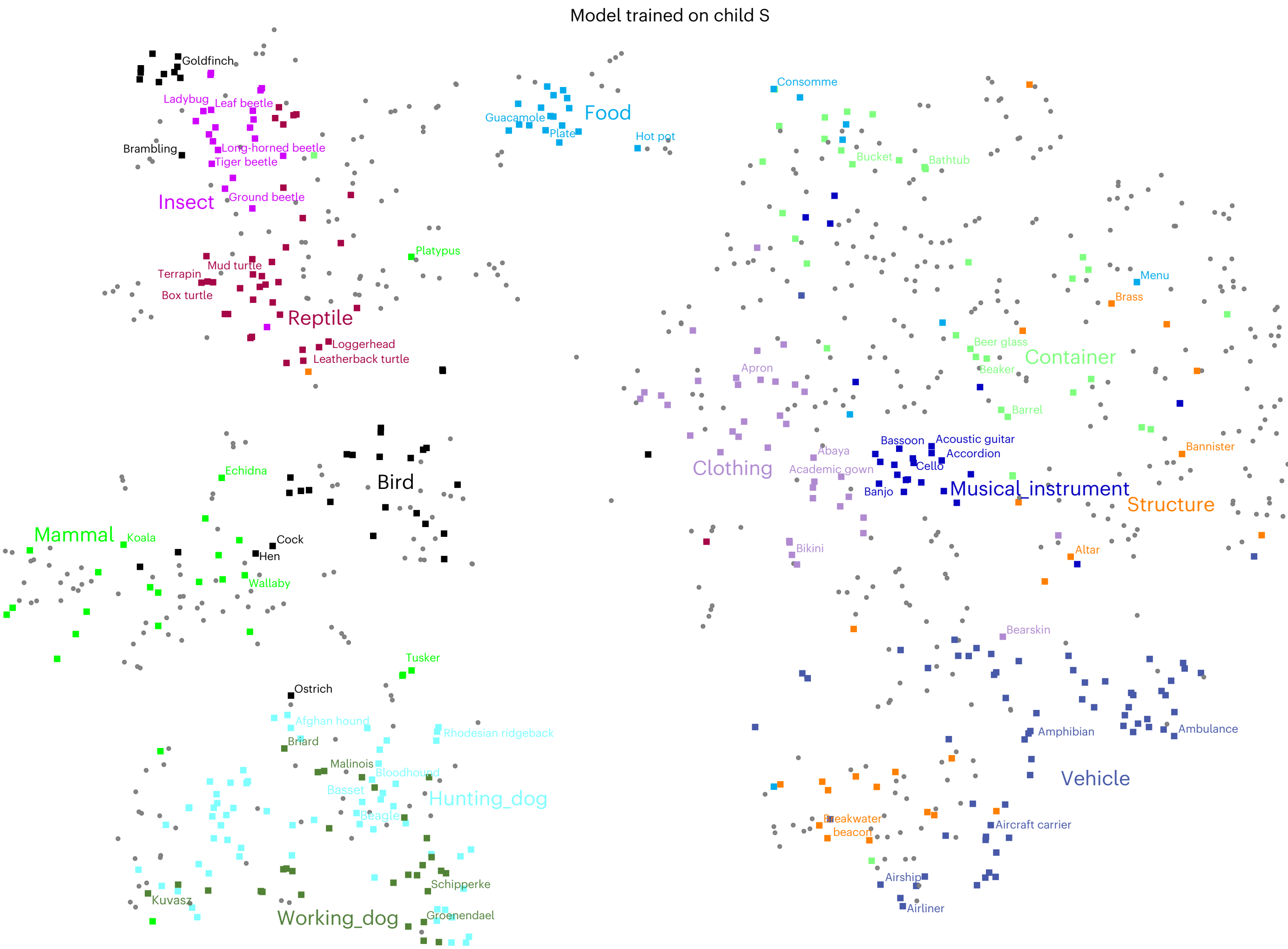

Fig. 4 | *t*-distributed stochastic neighbor embeddings of the ImageNet classes. The embeddings are obtained from a ViT-B/14 model trained with DINO on data from child S only. Each point corresponds to a different ImageNet class. The class embeddings are computed as the mean embedding over all validation images belonging to that class. Different colours represent 12 different super-classes (indicated in larger font) extracted from the WordNet hierarchy. Five classes are labelled individually for each super-class. For legibility, the other classes are not labelled individually. The visualizations for models trained on the other childrens' data are qualitatively very similar (Supplementary Figs. 3 and 4). More *t*-SNE visualizations can be found at the accompanying code repository.

of ImageNet consistently outperforms the SAYCam-trained models on images from the Konkle objects dataset, although the generation quality of SAYCam-trained models on this dataset can be improved substantially with a small amount of finetuning.

Discussion

In this article, we investigated what state-of-the-art SSL algorithms can learn from a sample of a child's longitudinal, egocentric visual experience without strong inductive biases. Our analyses reveal both strengths and weaknesses of the representations learned from a child's visual experience with current SSL algorithms. On the one hand, with the equivalent of a few weeks of visual experience only, models trained with data from individual children already perform at 65–70% of a high-performance ImageNet-trained model in a diverse range of downstream evaluation tasks (Fig. 2). They can also learn to localize semantic categories in an image without any explicit location supervision (Fig. 3a), and they can learn broad semantic categories in an unsupervised way (Fig. 4). Thus, despite substantial differences between the visual experience of a developing child and the standard datasets used for training state-of-the-art computer vision models[27], models trained with a realistic proxy of a child's visual experience still display highly non-trivial visual capabilities. These capabilities are also surprisingly consistent across models trained on different children in SAYCam (Fig. 2c; also see Supplementary Fig. 7), even with substantial individual differences in the environments and behaviours of these children[13]. On the other hand, these models seem to be less object-centric than models trained with large-scale, photographic image datasets like ImageNet (Fig. 3b), and in generative tests with out-of-domain stimuli, they seem to struggle with fine object details, even though they can successfully extrapolate the texture, colour, orientation and rough outlines of objects (Fig. 6).

In our experiments, we used reference models trained on different types of visual data to better situate the capabilities of the SAYCam-trained models. Some of these reference models display visual capabilities comparable to the models trained on individual children in SAYCam (Fig. 2c), for example, ImageNet (10%), Ego4D-200h, even Kinetics-200h to some extent, despite substantial differences between these visual data. This result suggests a considerable degree of robustness in the emergence of these general visual capabilities. Some earlier works, on the other hand, emphasized the special properties of child-centric visual data from a representation learning perspective[27–29]. Our results are not necessarily inconsistent with these studies;

https://doi.org/10.1038/s42256-024-00802-0

Fig. 5 | Nearest neighbours in the embedding space. a,**b**, The leftmost column shows six query images; the next ten images in each row are the ten nearest neighbours in the embedding space. Results from a ViT-B/14 DINO model trained on child S (**a**) and from a random, untrained model with the same architecture (**b**) are shown. Nearest neighbours are with respect to the Euclidean metric. Both the query and the nearest neighbours are from the Open Images V7 dataset[26]. Detailed image credits can be found in Supplementary Table 2.

because we focused on relatively broad measures of performance in our qualitative and quantitative evaluations, we cannot rule out more fine-grained differences between the models that might be hidden behind their comparable overall performance. However, isolating the causes of such potential fine-grained differences would be difficult in our case, as our reference datasets differ across many dimensions.

What are the implications of our results for the 'nature versus nurture' question regarding the acquisition of basic visual capabilities, such as real-world object recognition? Motivated by the early emergence of some visual capabilities in infants, developmental psychologists postulated various innate constraints related to objects, agents, space and categories[3,4,9–11], hypothesized to be critical for subsequent learning. However, a rigorous computational test of these claims requires considering both a sufficiently realistic proxy of a child's actual visual experience and powerful, generic, scalable learning algorithms and models. Arguably for the first time in history, we now have both ingredients, thanks to advances in the collection of large-scale longitudinal developmental datasets like SAYCam[13] and advances in deep learning, giving us powerful generic learning algorithms and architectures. Together with a handful of other recent studies[28,30–34], this work is

https://doi.org/10.1038/s42256-024-00802-0

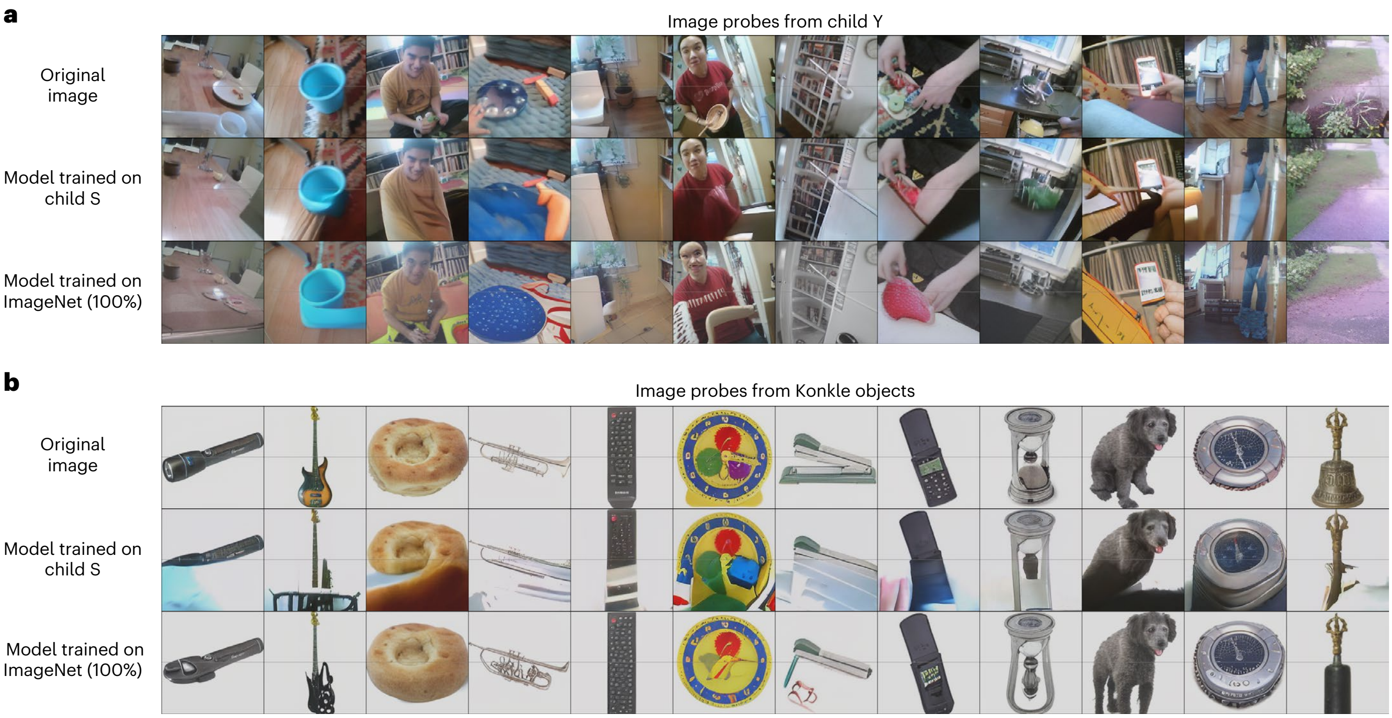

Fig. 6 | Qualitative evaluation of the generative models. a,**b**, Conditional samples from two different models (trained on child S in SAYCam or on all of ImageNet) seeded with images from child Y in SAYCam (**a**) or with images from the Konkle objects dataset (**b**). In each case, the upper half of the image is given to the model as context, and the lower half is generated by the model. All model completions are zero-shot (the models have not seen any prior examples from these datasets). More examples can be found at the accompanying code repository.

among the first to take advantage of these new opportunities to address fundamental questions in cognitive science. Our results, for example, suggest that strong inductive biases like a taxonomic generalization bias or an innate ability to segment objects may be unnecessary, as our generic self-supervised models already do a reasonably good job of learning to segment objects in images (Fig. 3a) or to categorize objects based on their kind (Fig. 4) from limited and noisy visual data available to a child without such inductive biases. However, the models' ability to cleanly segment and generate objects is imperfect (Figs. 3b and 6), so it remains an open empirical question if they can attain human-level understanding of objects by simply being trained on developmentally more realistic amounts of data or if stronger object-centric inductive biases may still be necessary to achieve this[3,4,35].

There are several differences between our experimental setting and the actual learning problem faced by children. These differences should be kept in mind when considering the implications of our results for developmental psychology. First, even the combined data from SAYCam amount to roughly 40 days of visual experience (factoring in 12 hours of sleep per day). To extend this to developmentally realistic amounts of data would require roughly two orders of magnitude more data than we currently have. The capabilities of the current models would undoubtedly improve with additional data at this scale even without any other changes, but it is an open empirical question how much they would improve. Second, here we only considered visual data, but a child's actual experience is multimodal, with auditory, haptic and sensorimotor components, in addition to vision. The capabilities of the current models would again likely improve with these complementary sources of information. Third, our models are trained with stochastic gradient descent, which is biologically implausible in the context of deep networks[36]. To the extent that biological learning must satisfy demanding constraints that are not relevant for deep learning, our results may overestimate what can be learned from a child's visual experience with biologically plausible learning mechanisms. Compared to deep learning models, this may necessitate more reliance on innate inductive biases in humans.

Another difference is that children are interactive learners. They learn their own behavioural policies regarding how to interact with objects or other agents in the environment. This allows them to shape and structure their own sensory experiences. Our models, on the other hand, are passive learners. The learnability results here thus relate to what is learnable from a visual stream that is, to some extent, already structured by the child. Interactive models that can actively shape their own experiences, as children do, might learn more effectively compared to passive learners[37], in which case our results would underestimate what can be learned from child-like visual experience without strong inductive biases.

There are also important differences between the raw visual inputs received by our models and those received by children. The SAYCam frames have relatively low spatial resolution (640 × 480 pixels) compared to the human retina. They contain a substantial amount of motion blur artefacts, and the image quality is generally poor in low lighting conditions. Efforts to collect higher quality headcam data with better cameras are already under way[38]. Modern SSL algorithms often use heavy data augmentation strategies like colour jittering or random resized cropping (the particular data augmentations used by each of our SSL algorithms are detailed in Methods). These augmentations increase the effective sample size to the benefit of the models. It is unclear whether similar processes in humans could implicitly expand the input in a biologically plausible way. Foveation represents an interesting example in this respect, with its functional similarity to random resized cropping.

We hope that our work will inspire new collaborations between machine learning and developmental psychology[27,29,39], as the impact of modern deep learning on developmental psychology has been relatively limited thus far. One key reason for this is the data gap[40] between machine and human learners; for example, today's computer vision models are typically trained with visual data that are very different in content, style and amount from a child's visual experience (for the models, millions, sometimes billions, of static photographic pictures scraped from the internet versus, for the child, a few years of continuous, egocentric data streams from the world). Here we bridged this data gap by training the same models on a realistic proxy of a child's egocentric visual experience and demonstrating these models' powerful visual capabilities. Future algorithmic advances, combined with richer and larger developmental datasets, can be evaluated through the same approach, further enriching our understanding of what can be learned from a child's experience with minimal inductive biases.

Methods

Evaluation tasks for the embedding models

Here we describe the nine tasks used for evaluating the embedding models, each associated with a dataset.

Labeled S. Labeled S contains ~58,000 manually labelled frames from child S in SAYCam[30]. We use the temporally ×10 subsampled version of this dataset (0.1 frames per second) containing ~5,800 images from 26 different classes. Temporal subsampling reduces the temporal correlations in the dataset and makes the classification task more challenging. We then randomly split the data in half, use the first half for training and the second half for evaluation. This is our only within-domain evaluation task for models trained on SAYCam, specifically for models trained on data from child S.

Konkle objects. This is a public dataset available from ref. 41. The images in this dataset depict common everyday objects in isolation against a uniform white background[42]. We only use a subset of the categories from the dataset that contains a sufficiently large number of exemplars, that is, 16 or 17 exemplars. This subset contains 4040 images from 240 different object categories. We split the data in half, use the first half for training and the second half for evaluation.

CORe50. This is a public dataset available from ref. 43. The dataset contains 50 different everyday objects undergoing various continuous transformations (complex combinations of 3D rotations and translations) against a variety of backgrounds[44]. The dataset is originally in video format, but we sample the videos at five frames per second to make an image dataset. Each object is shot against the same set of 11 unique backgrounds. We use six of these backgrounds for training and the remaining five backgrounds for evaluation (90,000 images in total for training, 75,000 images for evaluation). This task thus tests whether a model can (1) ignore the background and primarily respond to the foreground object instead and (2) generalize over continuous transformations. Note that a model primarily responding to the background would perform at near chance levels (2% top-1 accuracy) in this task, since the background does not have any predictive value for the object identity.

ImageNet. ImageNet (ILSVRC-2012) is a large and diverse dataset of high-quality images from the internet[19] and is a very popular benchmark for real-world visual object recognition. The dataset is publicly available from ref. 45. We use the standard training–validation split for this dataset, containing ~1,280,000 training images and 50,000 validation images from 1,000 semantic classes.

ImageNet OOD. To evaluate the robustness, or out-of-distribution (OOD) generalization capabilities, of the trained models, we also consider out-of-distribution versions of the ImageNet benchmark[46,47]. The ImageNet OOD benchmark contains 17 different out-of-distribution versions of ImageNet generated by applying various transformations to images from the ImageNet validation set. These include transformations such as taking the silhouettes of the objects in the image, stylizing the image, adding different types of noise to the image, changing the colours in the image, etc. For evaluation, we use the OOD accuracy metric, which is just the mean top-1 accuracy over all 17 out-of-distribution datasets[47]. This evaluation dataset is publicly available from ref. 48.

Ecoset. Ecoset can be thought of as an ecologically more realistic version of ImageNet containing images from 565 basic-level categories only, selected for their concreteness and frequency of usage in language[49]. The dataset comes with a standard training–validation split containing ~1,440,000 training images and 28,250 validation images, which we use for training and evaluation, respectively. The dataset is publicly available from ref. 50.

Places365. Because the SAYCam dataset contains examples of various scene categories (living room, dining room, kitchen, bathroom, playground, beach, street, porch, and so on) in addition to object categories, we are interested in evaluating the capacity of SAYCam-trained models to recognize places as well as objects. For this purpose, we use the Places365 dataset[51]. Places365 contains ~1,800,000 training images and 36,500 validation images from 365 different place categories. The dataset is publicly available from ref. 52.

DAVIS-2017. A good visual representation is ideally a general-purpose representation that can be used profitably not just in visual recognition tasks, but in a broader range of downstream tasks. For this reason, we also evaluate the SAYCam-learned representations in two dense prediction tasks. DAVIS-2017 is a video object segmentation task where the model is given a ground-truth segmentation mask for the initial frame of a short video clip and is expected to predict the segmentation masks for the following frames in the video[53]. In common evaluation protocols used for this task, the predicted segmentation masks for the non-initial frames are computed with a non-parametric message passing type algorithm that uses the representations of the frames and the predicted segmentation masks for nearby frames. This task essentially evaluates how robust the model's representations of the objects in the video clip are to spatiotemporal transformations that take place in the clip: more robust representations are expected to propagate the initial ground-truth segmentation masks better. The evaluation set consists of 30 video clips, each containing ~67 frames and ~2 objects on average. The data are publicly available from ref. 54.

COCO. We also evaluate our models on the semantic segmentation component of the COCO benchmark[55]. COCO is publicly available to download from ref. 56. Recall that in semantic segmentation the goal is to label each pixel of the image with the semantic category label of the object (or 'stuff') occupying that pixel. We use a subset of COCO that contains the 21 categories present in the Pascal VOC dataset. This subset has ~92,500 training images and 5,000 validation images in total.

For all evaluation tasks except DAVIS-2017 (including the COCO semantic segmentation task), we use linear readouts trained on top of frozen features, also known as a linear probe. For DAVIS-2017, as mentioned above, we use a standard non-parametric label propagation algorithm to predict the segmentation masks[57]. We use standard evaluation metrics for all our evaluation tasks: top-1 accuracy for the classification tasks, mean intersection over union for the COCO semantic segmentation task and the mean region and contour similarity for DAVIS-2017.

SSL algorithms for the embedding models

Here we describe each of the three SSL algorithms we used for training our embedding models. These algorithms represent a range of different modern approaches to self-supervised representation learning from static images or frames.

DINO. DINO is a self-distillation type representation learning algorithm[14], where a teacher model and a student model iteratively improve each other. During training, the teacher and the student

Table 1 | List of all trained embedding models (49 models in total)

Algorithms	Data	Models				
		ResNeXt-50	ViT-B/14	ViT-L/16	ViT-B/16	ViT-S/16
DINO	SAY	✓	✓	✓	✓	✓
	S	✓	✓	✓	✓	✓
	A	✓	✓	✓	✓	✓
	Y	✓	✓	✓	✓	✓
	Ego4D-200h		✓			
	Kinetics-200h		✓			
	ImageNet (100%)		✓			
	ImageNet (10%)		✓			
	ImageNet (1%)		✓			
Mugs	SAY			✓	✓	✓
	S			✓	✓	✓
	A			✓	✓	✓
	Y			✓	✓	✓
MAE	SAY			✓	✓	✓
	S			✓	✓	✓
	A			✓	✓	✓
	Y			✓	✓	✓

The trained combinations of algorithm, data and model are indicated by check marks.

receive different copies of the same image, transformed in various ways with a set of data augmentation methods, and the objective of the algorithm is to push the representations of these copies towards each other, because they share the same semantic content. The data augmentation methods used in DINO are colour jitter, random resized crops, horizontal flips, grey-scaling, Gaussian blur and solarization.

Mugs. Mugs is a hybrid SSL algorithm combining ideas from self-distillation and contrastive learning to learn multi-granular visual representations[15]. Mugs uses the same set of data augmentations as DINO.

Masked autoencoders. MAEs use reconstruction of masked image patches as the SSL objective[16]. By learning to predict masked patches from visible patches, the algorithm expects to learn higher level, semantically useful regularities in visual scenes (for example, learning that the face, the legs and the tail of a dog often appear in a particular configuration). MAEs use a much lighter data augmentation pipeline than other algorithms, requiring only random resized crops and horizontal flips. As recommended[16], we use a large masking ratio of 75% during training, that is, 75% of the image patches are randomly masked out.

We generally use the default hyperparameter choices and training configurations recommended for these algorithms in the original papers, with minor modifications. We use the same data augmentation pipeline for every model trained with a given algorithm. Further details can be found in the corresponding training codes that can be accessed from our main public repository.

Reference datasets for the embedding models

Kinetics-700 consists of short YouTube clips of people performing various actions, representing 700 different action categories[20]. Kinetics-700 is publicly available for download from ref. 58. The video clips in Kinetics-700 are typically shorter than ten seconds, hence the dataset overall is expected to be much more diverse in style and content and temporally much less correlated than SAYCam. Ego4D, on the other hand, has more similar temporal characteristics to SAYCam; the videos are temporally extended, continuous, egocentric headcam recordings, with recording sessions lasting tens of minutes on average[21]. The main differences from SAYCam are (1) the videos are taken from the perspective of adult camera wearers, not from the perspective of young children, and (2) the recordings are made by many more individuals than the SAYCam recordings. In Ego4D, each individual contributes ~4 hours of recording on average, so a 200-hour-long subset of the dataset would be expected to contain recordings from roughly 50 different camera wearers, in contrast to a single child in SAYCam. Ego4D is publicly available from ref. 59 (after signing a license agreement). We use 200-hour-long subsets of these datasets, because 200 hours is roughly equal to the total length of the video data we have available from one of the children in SAYCam, namely S. To obtain these 200-hour long subsets, we use the first 128 clips from each class in Kinetics-700 and select a continuous chunk of videos from Ego4D with a random starting point until the total length of the videos in the selection roughly equals 200 hours.

Training details for the embedding models

We train each model for four days on four A100 graphics processing units (GPUs) with 80 GB GPU memory, using data parallelism (the ViT-B/14 DINO model trained on all of ImageNet was trained for four additional days to make sure it was not under-trained). We use the Adam optimizer to train all models[60]. In each experiment, we use either a batch size of 512 or the largest batch size we could fit on four GPUs, in those cases where we could not fit a total batch size of 512 on the GPUs. Batch sizes and learning rates thus vary across experiments. Inspection of the training losses confirms that they all saturate, hence under-training is unlikely for any of our pretraining runs (all training logs are made available in our public repository). Table 1 presents a concise list of all embedding models trained for this work.

Class activation maps

In visualizing the CAMs shown in Fig. 3a, we first normalize the linearly combined and upsampled feature map to have zero mean and unit variance, where the mean and variance are estimated over a batch of images

Table 2 | List of all trained generative models (23 models in total)

Pretraining data	Finetuning data		
	Konkle (iid)	Konkle (non-vehicle)	None
SAY	✓	✓	✓
S	✓	✓	✓
A	✓	✓	✓
Y	✓	✓	✓
ImageNet (100%)	✓	✓	✓
ImageNet (10%)	✓	✓	✓
ImageNet (1%)	✓	✓	✓
None	✓	✓	

The trained combinations of pretraining and finetuning data are indicated by check marks. 'None' means pretraining (or finetuning) was not applied.

from the same class, pass the normalized map through a pointwise sigmoid nonlinearity and then scale it by 255 so that the values in the final map are between 0 and 255 (or, in torch notation: m = 255 * torch.sigmoid((m-torch.mean(m))/torch.std(m)). We then alpha-blend this activation map with the original image using a blending coefficient of 0.8 for the map and 0.2 for the image.

Additional details about the generative models

We train customized VQGAN models using the Taming Transformers repository made available by the authors of VQGAN[22]. The Taming Transformers repository can be accessed at ref. 61. For the GPT model, we use a standard 730 million-parameter GPT model that is similar to OpenAI's gpt2-large model[23]. Using the same architecture, we also train reference VQGAN-GPT models on ImageNet, using either 100%, 10%, or 1% of the training set, as described previously.

For the VQGAN component of the generative models for SAYCam, we use a codebook with a vocabulary size of 8,192 and a spatial resolution of 32×32 (thus each frame is encoded as a 32×32 grid of integers, where the integers take values between 1 and 8,192). For the encoded SAYCam frames, the spatial resolution of 32×32 corresponds to a sequence length of 1,024 tokens. Due to computational constraints, the VQGAN models for ImageNet use a spatial resolution of 16×16 and a codebook with a dictionary size of 16,384. To train the VQGAN component of the generative model, we use the Taming Transformers repository (model configuration files are available from our public repository). The GPT component of the generative models has 36 layers, 20 attention heads and an embedding dimensionality of 1,280 in all cases (the model configuration is equivalent to OpenAI's gpt2-large model). We generate the model completions through exact sampling, with the softmax temperature set to $T = 1.0$.

Training and evaluation details for the generative models

SAYCam-trained GPT models were trained for four days on 16 A100 GPUs with a batch size of 96 (the model trained on the combined data from SAYCam was trained for four additional days to make sure it was not under-trained). The training logs (all made available from our public repository) confirm that under-training is not a serious concern for any of our models. The ImageNet-trained models were trained on eight A100 GPUs with a total batch size of 256 (the model trained on 100% of ImageNet was trained for 6 days, whereas the models trained on 10% and 1% of ImageNet were trained for 2 days only due to the more limited size of the training data in these cases). All models were trained with the Adam algorithm. Table 2 presents a concise list of all generative models trained for this work.

We measure the overall quality of the completions with the FID between the model generated samples and the ground-truth images[62]. We use three different image completion tasks to quantitatively evaluate the generative models: Labeled S, Konkle independent-identically-distributed (iid) and Konkle out-of-distribution (ood). In Labeled S, we use images from the validation split of the Labeled S dataset described above for the image completion task. In Konkle-iid, we randomly split the Konkle objects dataset in half, use the first half for training or finetuning the generative models and use the other half for the image completion task. In Konkle-ood, we split the Konkle objects dataset into non-overlapping vehicle and non-vehicle categories, use the non-vehicle categories for training or finetuning the generative models and use the vehicle categories (144 images in total) for the image completion task. Since this is an OOD generalization task, it is expected to be more challenging than the iid condition. The results are presented in Supplementary Table 1, which shows the FID scores of different models in each image completion task.

Reporting summary

Further information on research design is available in the Nature Portfolio Reporting Summary linked to this article.

Data availability

Except for SAYCam, all data used in this study are publicly available. Instructions for accessing the public datasets are detailed in Methods. The SAYCam dataset can be accessed by authorized users with an institutional affiliation from the following Databrary repository: https://doi.org/10.17910/b7.564. The 'Labeled S' evaluation dataset, which is a subset of SAYCam, is also available from the same repository under the session name 'Labeled S'.

Code availability

All of our pretrained models (over 70 different models), as well as a variety of tools to use and analyse them, are available from the following public repository: https://github.com/eminorhan/silicon-menagerie (ref. 63). The repository also contains further examples of (1) attention and class activation maps, (2) *t*-SNE visualizations of embeddings, (3) nearest neighbour retrievals from the embedding models and (4) unconditional and conditional samples from the generative models. The code used for training and evaluating all the models is also publicly available from the same repository.

References

1. Bomba, P. & Siqueland, E. The nature and structure of infant form categories. *J. Exp. Child Psychol.* **35**, 294–328 (1983).
2. Murphy, G. *The Big Book of Concepts* (MIT, 2002).
3. Kellman, P. & Spelke, E. Perception of partly occluded objects in infancy. *Cogn. Psychol.* **15**, 483–524 (1983).
4. Spelke, E., Breinlinger, K., Macomber, J. & Jacobson, K. Origin of knowledge. *Psychol. Rev.* **99**, 605–632 (1992).
5. Ayzenberg, V. & Lourenco, S. Young children outperform feed-forward and recurrent neural networks on challenging object recognition tasks. *J. Vis.* **20**, 310–310 (2020).
6. Huber, L. S., Geirhos, R. & Wichmann, F. A. The developmental trajectory of object recognition robustness: children are like small adults but unlike big deep neural networks. *J. Vis.* **23**, 4 (2023).
7. Locke, J. *An Essay Concerning Human Understanding* (ed. Fraser, A. C.) (Clarendon Press, 1894).
8. Leibniz, G. *New Essays on Human Understanding* 2nd edn (eds Remnant, P. & Bennett, J.) (Cambridge Univ. Press, 1996).
9. Spelke, E. Initial knowledge: six suggestions. *Cognition* **50**, 431–445 (1994).
10. Markman, E. *Categorization and Naming in Children* (MIT, 1989).
11. Merriman, W., Bowman, L. & MacWhinney, B. The mutual exclusivity bias in children's word learning. *Monogr. Soc. Res. Child Dev.* **54**, 1–132 (1989).
12. Elman, J., Bates, E. & Johnson, M. *Rethinking Innateness: A Connectionist Perspective on Development* (MIT, 1996).

13. Sullivan, J., Mei, M., Perfors, A., Wojcik, E. & Frank, M. SAYCam: a large, longitudinal audiovisual dataset recorded from the infant's perspective. *Open Mind* **5**, 20–29 (2022).
14. Caron, M. et al. Emerging properties in self-supervised vision transformers. In *Proc. IEEE/CVF International Conference on Computer Vision* 9650–9660 (IEEE, 2021).
15. Zhou, P. et al. Mugs: a multi-granular self-supervised learning framework. Preprint at https://arxiv.org/abs/2203.14415 (2022).
16. He, K. et al. Masked autoencoders are scalable vision learners. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition* 15979–15988 (IEEE, 2022).
17. Dosovitskiy, A. et al. An image is worth 16x16 words: transformers for image recognition at scale. In *International Conference on Learning Representations* (2020).
18. Xie, S., Girshick, R., Dollár, P., Tu, Z. & He, K. Aggregated residual transformations for deep neural networks. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition* 1492–1500 (IEEE, 2017).
19. Russakovsky, O. et al. ImageNet large scale visual recognition challenge. *Int. J. Comput. Vis.* **115**, 211–252 (2015).
20. Smaira, L. et al. A short note on the Kinetics-700-2020 human action dataset. Preprint at https://arxiv.org/abs/2010.10864 (2020).
21. Grauman, K. et al. Ego4D: around the world in 3,000 hours of egocentric video. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 18995–19012 (IEEE, 2022).
22. Esser, P., Rombach, R. & Ommer, B. Taming transformers for high-resolution image synthesis. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition* 12873–12883 (IEEE, 2021).
23. Radford, A. et al. Language models are unsupervised multitask learners. *OpenAI* https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf (2019).
24. Zhou, B., Khosla, A., Lapedriza, A., Oliva, A. & Torralba, A. Learning deep features for discriminative localization. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition* 2921–2929 (IEEE, 2016).
25. van der Maaten, L. & Hinton, G. Visualizing data using t-SNE. *J. Mach. Learn. Res.* **9**, 2579–2605 (2008).
26. Kuznetsova, A. et al. The Open Images Dataset V4. *Int. J. Comput. Vis.* **128**, 1956–1981 (2020).
27. Smith, L. & Slone, L. A developmental approach to machine learning? *Front. Psychol.* **8**, 2124 (2017).
28. Bambach, S., Crandall, D., Smith, L. & Yu, C. Toddler-inspired visual object learning. *Adv. Neural Inf. Process. Syst.* **31**, 1209–1218 (2018).
29. Zaadnoordijk, L., Besold, T. & Cusack, R. Lessons from infant learning for unsupervised machine learning. *Nat. Mach. Intell.* **4**, 510–520 (2022).
30. Orhan, E., Gupta, V. & Lake, B. Self-supervised learning through the eyes of a child. *Adv. Neur. In.* **33**, 9960–9971 (2020).
31. Lee, D., Gujarathi, P. & Wood, J. Controlled-rearing studies of newborn chicks and deep neural networks. Preprint at https://arxiv.org/abs/2112.06106 (2021).
32. Zhuang, C. et al. Unsupervised neural network models of the ventral visual stream. *Proc. Natl Acad. Sci. USA* **118**, e2014196118 (2021).
33. Zhuang, C. et al. How well do unsupervised learning algorithms model human real-time and life-long learning? In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track* (2022).
34. Vong, W. K., Wang, W., Orhan, A. E. & Lake, B. M. Grounded language acquisition through the eyes and ears of a single child. *Science* **383**, 504–511 (2024).
35. Locatello, F. et al. Object-centric learning with slot attention. *Adv. Neur. In.* **33**, 11525–11538 (2020).
36. Lillicrap, T., Santoro, A., Marris, L., Akerman, C. & Hinton, G. Backpropagation and the brain. *Nat. Rev. Neurosci.* **21**, 335–346 (2020).
37. Gureckis, T. & Markant, D. Self-directed learning: a cognitive and computational perspective. *Perspect. Psychol. Sci.* **7**, 464–481 (2012).
38. Long, B. et al. The BabyView camera: designing a new head-mounted camera to capture children's early social and visual environments. *Behav. Res. Methods* https://doi.org/10.3758/s13428-023-02206-1 (2023).
39. Moore, D., Oakes, L., Romero, V. & McCrink, K. Leveraging developmental psychology to evaluate artificial intelligence. In *2022 IEEE International Conference on Development and Learning (ICDL)* 36–41 (IEEE, 2022).
40. Frank, M. C. Bridging the data gap between children and large language models. *Trends Cogn. Sci.* **27**, 990–992 (2023).
41. Object stimuli. *Brady Lab* https://bradylab.ucsd.edu/stimuli/ObjectCategories.zip
42. Konkle, T., Brady, T., Alvarez, G. & Oliva, A. Conceptual distinctiveness supports detailed visual long-term memory for real-world objects. *J. Exp. Psychol. Gen.* **139**, 558 (2010).
43. Lomonaco, V. & Maltoni, D. CORe50 Dataset. *GitHub* https://vlomonaco.github.io/core50 (2017).
44. Lomonaco, V. & Maltoni, D. CORe50: a new dataset and benchmark for continuous object recognition. In *Proc. 1st Annual Conference on Robot Learning* (eds Levine, S. et al.) 17–26 (PMLR, 2017).
45. Russakovsky, O. et al. ImageNet Dataset. https://www.image-net.org/download.php (2015).
46. Geirhos, R. et al. Shortcut learning in deep neural networks. *Nat. Mach. Intell.* **2**, 665–673 (2020).
47. Geirhos, R. et al. Partial success in closing the gap between human and machine vision. *Adv. Neur. In.* **34**, 23885–23899 (2021).
48. Geirhos, R. et al. ImageNet OOD Dataset. *GitHub* https://github.com/bethgelab/model-vs-human (2021).
49. Mehrer, J., Spoerer, C., Jones, E., Kriegeskorte, N. & Kietzmann, T. An ecologically motivated image dataset for deep learning yields better models of human vision. *Proc. Natl Acad. Sci. USA* **118**, e2011417118 (2021).
50. Mehrer, J., Spoerer, C., Jones, E., Kriegeskorte, N. & Kietzmann, T. Ecoset Dataset. *Hugging Face* https://huggingface.co/datasets/kietzmannlab/ecoset (2021).
51. Zhou, B., Lapedriza, A., Khosla, A., Oliva, A. & Torralba, A. Places: a 10 million image database for scene recognition. *IEEE T. Pattern Anal.* **40**, 1452–1464 (2017).
52. Zhou, B. et al. Places365 Dataset. http://places2.csail.mit.edu (2017).
53. Pont-Tuset, J. et al. The 2017 DAVIS challenge on video object segmentation. Preprint at https://arxiv.org/abs/1704.00675 (2017).
54. Pont-Tuset, J. et al. DAVIS-2017 evaluation code, dataset and results. https://davischallenge.org/davis2017/code.html (2017).
55. Lin, T. et al. Microsoft COCO: common objects in context. In *Computer Vision – ECCV 2014* (eds Fleet, D. et al.) 740–755 (2014).
56. COCO Dataset. https://cocodataset.org/#download (2014).
57. Jabri, A., Owens, A. & Efros, A. Space-time correspondence as a contrastive random walk. *Adv. Neur. In.* **33**, 19545–19560 (2020).
58. Kinetics-700-2020 Dataset. https://github.com/cvdfoundation/kinetics-dataset#kinetics-700-2020 (2020).
59. Ego4D Dataset. https://ego4d-data.org/ (2022).
60. Kingma, D. & Ba, J. Adam: A method for stochastic optimization. Preprint at https://arxiv.org/abs/1412.6980 (2014).
61. VQGAN resources. GitHub https://github.com/CompVis/taming-transformers (2021).
62. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B. & Hochreiter, S. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. *Adv. Neural Inf. Process. Syst.* **30**, 6629–6640 (2017).
63. Orhan, A. E. eminorhan/silicon-menagerie: v1.0.0-alpha. *Zenodo* https://doi.org/10.5281/zenodo.8322408 (2023).

Acknowledgements

We thank W. K. Vong, A. Tartaglini and M. Ren for helpful discussions and comments on an earlier version of this paper. This work was supported by the DARPA Machine Common Sense program (B.M.L.) and NSF Award 1922658 NRT-HDR: FUTURE Foundations, Translation and Responsibility for Data Science (B.M.L.).

Author contributions

A.E.O. and B.M.L. conceptualized and designed the study. A.E.O. implemented the experiments. A.E.O. analysed the results with feedback from B.M.L. A.E.O. wrote the first draft. B.M.L. reviewed and edited the paper.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s42256-024-00802-0.

Correspondence and requests for materials should be addressed to A. Emin Orhan.

Peer review information *Nature Machine Intelligence* thanks Rhodri Cusack, Cliona O'Doherty, Masataka Sawayama and the other, anonymous, reviewer(s) for their contribution to the peer review of this work.

nature portfolio

Corresponding author(s): Emin Orhan

Last updated by author(s): December 25, 2023

Reporting Summary

Nature Portfolio wishes to improve the reproducibility of the work that we publish. This form provides structure for consistency and transparency in reporting. For further information on Nature Portfolio policies, see our Editorial Policies and the Editorial Policy Checklist.

Statistics

For all statistical analyses, confirm that the following items are present in the figure legend, table legend, main text, or Methods section.

n/a	Confirmed	
☐	☒	The exact sample size (*n*) for each experimental group/condition, given as a discrete number and unit of measurement
☐	☒	A statement on whether measurements were taken from distinct samples or whether the same sample was measured repeatedly
☒	☐	The statistical test(s) used AND whether they are one- or two-sided *Only common tests should be described solely by name; describe more complex techniques in the Methods section.*
☒	☐	A description of all covariates tested
☒	☐	A description of any assumptions or corrections, such as tests of normality and adjustment for multiple comparisons
☐	☒	A full description of the statistical parameters including central tendency (e.g. means) or other basic estimates (e.g. regression coefficient) AND variation (e.g. standard deviation) or associated estimates of uncertainty (e.g. confidence intervals)
☒	☐	For null hypothesis testing, the test statistic (e.g. F, t, r) with confidence intervals, effect sizes, degrees of freedom and P value noted *Give P values as exact values whenever suitable.*
☒	☐	For Bayesian analysis, information on the choice of priors and Markov chain Monte Carlo settings
☒	☐	For hierarchical and complex designs, identification of the appropriate level for tests and full reporting of outcomes
☒	☐	Estimates of effect sizes (e.g. Cohen's d, Pearson's r), indicating how they were calculated

Our web collection on statistics for biologists contains articles on many of the points above.

Software and code

Policy information about availability of computer code

Data collection	no software was used for data collection
Data analysis	Python 3.10, Numpy 1.23

For manuscripts utilizing custom algorithms or software that are central to the research but not yet described in published literature, software must be made available to editors and reviewers. We strongly encourage code deposition in a community repository (e.g. GitHub). See the Nature Portfolio guidelines for submitting code & software for further information.

Data

Policy information about availability of data

All manuscripts must include a data availability statement. This statement should provide the following information, where applicable:

- Accession codes, unique identifiers, or web links for publicly available datasets
- A description of any restrictions on data availability
- For clinical datasets or third party data, please ensure that the statement adheres to our policy

A Data availability statement is provided in the manuscript.

Human research participants

Policy information about studies involving human research participants and Sex and Gender in Research.

Reporting on sex and gender	*Use the terms sex (biological attribute) and gender (shaped by social and cultural circumstances) carefully in order to avoid confusing both terms. Indicate if findings apply to only one sex or gender; describe whether sex and gender were considered in study design whether sex and/or gender was determined based on self-reporting or assigned and methods used. Provide in the source data disaggregated sex and gender data where this information has been collected, and consent has been obtained for sharing of individual-level data; provide overall numbers in this Reporting Summary. Please state if this information has not been collected. Report sex- and gender-based analyses where performed, justify reasons for lack of sex- and gender-based analysis.*
Population characteristics	*Describe the covariate-relevant population characteristics of the human research participants (e.g. age, genotypic information, past and current diagnosis and treatment categories). If you filled out the behavioural & social sciences study design questions and have nothing to add here, write "See above."*
Recruitment	*Describe how participants were recruited. Outline any potential self-selection bias or other biases that may be present and how these are likely to impact results.*
Ethics oversight	IRB approval from NYU for using the SAYCam dataset: IRB-FY2018-2143. Additional details regarding the SAYCam dataset can be found in the original SAYCam paper cited in our manuscript (Sullivan et al., Open Mind, 2022), since the data was not collected by us.

Note that full information on the approval of the study protocol must also be provided in the manuscript.

Field-specific reporting

Please select the one below that is the best fit for your research. If you are not sure, read the appropriate sections before making your selection.

☐ Life sciences ☐ Behavioural & social sciences ☐ Ecological, evolutionary & environmental sciences

For a reference copy of the document with all sections, see nature.com/documents/nr-reporting-summary-flat.pdf

Life sciences study design

All studies must disclose on these points even when the disclosure is negative.

Sample size	*Describe how sample size was determined, detailing any statistical methods used to predetermine sample size OR if no sample-size calculation was performed, describe how sample sizes were chosen and provide a rationale for why these sample sizes are sufficient.*
Data exclusions	*Describe any data exclusions. If no data were excluded from the analyses, state so OR if data were excluded, describe the exclusions and the rationale behind them, indicating whether exclusion criteria were pre-established.*
Replication	*Describe the measures taken to verify the reproducibility of the experimental findings. If all attempts at replication were successful, confirm this OR if there are any findings that were not replicated or cannot be reproduced, note this and describe why.*
Randomization	*Describe how samples/organisms/participants were allocated into experimental groups. If allocation was not random, describe how covariates were controlled OR if this is not relevant to your study, explain why.*
Blinding	*Describe whether the investigators were blinded to group allocation during data collection and/or analysis. If blinding was not possible, describe why OR explain why blinding was not relevant to your study.*

Behavioural & social sciences study design

All studies must disclose on these points even when the disclosure is negative.

Study description	*Briefly describe the study type including whether data are quantitative, qualitative, or mixed-methods (e.g. qualitative cross-sectional, quantitative experimental, mixed-methods case study).*
Research sample	*State the research sample (e.g. Harvard university undergraduates, villagers in rural India) and provide relevant demographic information (e.g. age, sex) and indicate whether the sample is representative. Provide a rationale for the study sample chosen. For studies involving existing datasets, please describe the dataset and source.*
Sampling strategy	*Describe the sampling procedure (e.g. random, snowball, stratified, convenience). Describe the statistical methods that were used to predetermine sample size OR if no sample-size calculation was performed, describe how sample sizes were chosen and provide a*

	rationale for why these sample sizes are sufficient. For qualitative data, please indicate whether data saturation was considered, and what criteria were used to decide that no further sampling was needed.
Data collection	*Provide details about the data collection procedure, including the instruments or devices used to record the data (e.g. pen and paper, computer, eye tracker, video or audio equipment) whether anyone was present besides the participant(s) and the researcher, and whether the researcher was blind to experimental condition and/or the study hypothesis during data collection.*
Timing	*Indicate the start and stop dates of data collection. If there is a gap between collection periods, state the dates for each sample cohort.*
Data exclusions	*If no data were excluded from the analyses, state so OR if data were excluded, provide the exact number of exclusions and the rationale behind them, indicating whether exclusion criteria were pre-established.*
Non-participation	*State how many participants dropped out/declined participation and the reason(s) given OR provide response rate OR state that no participants dropped out/declined participation.*
Randomization	*If participants were not allocated into experimental groups, state so OR describe how participants were allocated to groups, and if allocation was not random, describe how covariates were controlled.*

Ecological, evolutionary & environmental sciences study design

All studies must disclose on these points even when the disclosure is negative.

Study description	*Briefly describe the study. For quantitative data include treatment factors and interactions, design structure (e.g. factorial, nested, hierarchical), nature and number of experimental units and replicates.*
Research sample	*Describe the research sample (e.g. a group of tagged Passer domesticus, all Stenocereus thurberi within Organ Pipe Cactus National Monument), and provide a rationale for the sample choice. When relevant, describe the organism taxa, source, sex, age range and any manipulations. State what population the sample is meant to represent when applicable. For studies involving existing datasets, describe the data and its source.*
Sampling strategy	*Note the sampling procedure. Describe the statistical methods that were used to predetermine sample size OR if no sample-size calculation was performed, describe how sample sizes were chosen and provide a rationale for why these sample sizes are sufficient.*
Data collection	*Describe the data collection procedure, including who recorded the data and how.*
Timing and spatial scale	*Indicate the start and stop dates of data collection, noting the frequency and periodicity of sampling and providing a rationale for these choices. If there is a gap between collection periods, state the dates for each sample cohort. Specify the spatial scale from which the data are taken*
Data exclusions	*If no data were excluded from the analyses, state so OR if data were excluded, describe the exclusions and the rationale behind them, indicating whether exclusion criteria were pre-established.*
Reproducibility	*Describe the measures taken to verify the reproducibility of experimental findings. For each experiment, note whether any attempts to repeat the experiment failed OR state that all attempts to repeat the experiment were successful.*
Randomization	*Describe how samples/organisms/participants were allocated into groups. If allocation was not random, describe how covariates were controlled. If this is not relevant to your study, explain why.*
Blinding	*Describe the extent of blinding used during data acquisition and analysis. If blinding was not possible, describe why OR explain why blinding was not relevant to your study.*

Did the study involve field work? ☐ Yes ☐ No

Field work, collection and transport

Field conditions	*Describe the study conditions for field work, providing relevant parameters (e.g. temperature, rainfall).*
Location	*State the location of the sampling or experiment, providing relevant parameters (e.g. latitude and longitude, elevation, water depth).*
Access & import/export	*Describe the efforts you have made to access habitats and to collect and import/export your samples in a responsible manner and in compliance with local, national and international laws, noting any permits that were obtained (give the name of the issuing authority, the date of issue, and any identifying information).*
Disturbance	*Describe any disturbance caused by the study and how it was minimized.*

Reporting for specific materials, systems and methods

We require information from authors about some types of materials, experimental systems and methods used in many studies. Here, indicate whether each material, system or method listed is relevant to your study. If you are not sure if a list item applies to your research, read the appropriate section before selecting a response.

Materials & experimental systems

n/a	Involved in the study
☒	☐ Antibodies
☒	☐ Eukaryotic cell lines
☒	☐ Palaeontology and archaeology
☒	☐ Animals and other organisms
☒	☐ Clinical data
☒	☐ Dual use research of concern

Methods

n/a	Involved in the study
☒	☐ ChIP-seq
☒	☐ Flow cytometry
☒	☐ MRI-based neuroimaging

Antibodies

Antibodies used	*Describe all antibodies used in the study; as applicable, provide supplier name, catalog number, clone name, and lot number.*
Validation	*Describe the validation of each primary antibody for the species and application, noting any validation statements on the manufacturer's website, relevant citations, antibody profiles in online databases, or data provided in the manuscript.*

Eukaryotic cell lines

Policy information about cell lines and Sex and Gender in Research

Cell line source(s)	*State the source of each cell line used and the sex of all primary cell lines and cells derived from human participants or vertebrate models.*
Authentication	*Describe the authentication procedures for each cell line used OR declare that none of the cell lines used were authenticated.*
Mycoplasma contamination	*Confirm that all cell lines tested negative for mycoplasma contamination OR describe the results of the testing for mycoplasma contamination OR declare that the cell lines were not tested for mycoplasma contamination.*
Commonly misidentified lines (See ICLAC register)	*Name any commonly misidentified cell lines used in the study and provide a rationale for their use.*

Palaeontology and Archaeology

Specimen provenance	*Provide provenance information for specimens and describe permits that were obtained for the work (including the name of the issuing authority, the date of issue, and any identifying information). Permits should encompass collection and, where applicable, export.*
Specimen deposition	*Indicate where the specimens have been deposited to permit free access by other researchers.*
Dating methods	*If new dates are provided, describe how they were obtained (e.g. collection, storage, sample pretreatment and measurement), where they were obtained (i.e. lab name), the calibration program and the protocol for quality assurance OR state that no new dates are provided.*

☐ Tick this box to confirm that the raw and calibrated dates are available in the paper or in Supplementary Information.

Ethics oversight	*Identify the organization(s) that approved or provided guidance on the study protocol, OR state that no ethical approval or guidance was required and explain why not.*

Note that full information on the approval of the study protocol must also be provided in the manuscript.

Animals and other research organisms

Policy information about studies involving animals; ARRIVE guidelines recommended for reporting animal research, and Sex and Gender in Research

Laboratory animals	*For laboratory animals, report species, strain and age OR state that the study did not involve laboratory animals.*

Wild animals	*Provide details on animals observed in or captured in the field; report species and age where possible. Describe how animals were caught and transported and what happened to captive animals after the study (if killed, explain why and describe method; if released, say where and when) OR state that the study did not involve wild animals.*
Reporting on sex	*Indicate if findings apply to only one sex; describe whether sex was considered in study design, methods used for assigning sex. Provide data disaggregated for sex where this information has been collected in the source data as appropriate; provide overall numbers in this Reporting Summary. Please state if this information has not been collected. Report sex-based analyses where performed, justify reasons for lack of sex-based analysis.*
Field-collected samples	*For laboratory work with field-collected samples, describe all relevant parameters such as housing, maintenance, temperature, photoperiod and end-of-experiment protocol OR state that the study did not involve samples collected from the field.*
Ethics oversight	*Identify the organization(s) that approved or provided guidance on the study protocol, OR state that no ethical approval or guidance was required and explain why not.*

Note that full information on the approval of the study protocol must also be provided in the manuscript.

Clinical data

Policy information about clinical studies

All manuscripts should comply with the ICMJE guidelines for publication of clinical research and a completed CONSORT checklist must be included with all submissions.

Clinical trial registration	*Provide the trial registration number from ClinicalTrials.gov or an equivalent agency.*
Study protocol	*Note where the full trial protocol can be accessed OR if not available, explain why.*
Data collection	*Describe the settings and locales of data collection, noting the time periods of recruitment and data collection.*
Outcomes	*Describe how you pre-defined primary and secondary outcome measures and how you assessed these measures.*

Dual use research of concern

Policy information about dual use research of concern

Hazards

Could the accidental, deliberate or reckless misuse of agents or technologies generated in the work, or the application of information presented in the manuscript, pose a threat to:

No	Yes	
☒	☐	Public health
☒	☐	National security
☒	☐	Crops and/or livestock
☒	☐	Ecosystems
☒	☐	Any other significant area

Experiments of concern

Does the work involve any of these experiments of concern:

No	Yes	
☒	☐	Demonstrate how to render a vaccine ineffective
☒	☐	Confer resistance to therapeutically useful antibiotics or antiviral agents
☒	☐	Enhance the virulence of a pathogen or render a nonpathogen virulent
☒	☐	Increase transmissibility of a pathogen
☒	☐	Alter the host range of a pathogen
☒	☐	Enable evasion of diagnostic/detection modalities
☒	☐	Enable the weaponization of a biological agent or toxin
☒	☐	Any other potentially harmful combination of experiments and agents

ChIP-seq

Data deposition

- [ ] Confirm that both raw and final processed data have been deposited in a public database such as GEO.
- [ ] Confirm that you have deposited or provided access to graph files (e.g. BED files) for the called peaks.

Data access links *May remain private before publication.*	*For "Initial submission" or "Revised version" documents, provide reviewer access links. For your "Final submission" document, provide a link to the deposited data.*
Files in database submission	*Provide a list of all files available in the database submission.*
Genome browser session (e.g. UCSC)	*Provide a link to an anonymized genome browser session for "Initial submission" and "Revised version" documents only, to enable peer review. Write "no longer applicable" for "Final submission" documents.*

Methodology

Replicates	*Describe the experimental replicates, specifying number, type and replicate agreement.*
Sequencing depth	*Describe the sequencing depth for each experiment, providing the total number of reads, uniquely mapped reads, length of reads and whether they were paired- or single-end.*
Antibodies	*Describe the antibodies used for the ChIP-seq experiments; as applicable, provide supplier name, catalog number, clone name, and lot number.*
Peak calling parameters	*Specify the command line program and parameters used for read mapping and peak calling, including the ChIP, control and index files used.*
Data quality	*Describe the methods used to ensure data quality in full detail, including how many peaks are at FDR 5% and above 5-fold enrichment.*
Software	*Describe the software used to collect and analyze the ChIP-seq data. For custom code that has been deposited into a community repository, provide accession details.*

Flow Cytometry

Plots

Confirm that:

- [ ] The axis labels state the marker and fluorochrome used (e.g. CD4-FITC).
- [ ] The axis scales are clearly visible. Include numbers along axes only for bottom left plot of group (a 'group' is an analysis of identical markers).
- [ ] All plots are contour plots with outliers or pseudocolor plots.
- [ ] A numerical value for number of cells or percentage (with statistics) is provided.

Methodology

Sample preparation	*Describe the sample preparation, detailing the biological source of the cells and any tissue processing steps used.*
Instrument	*Identify the instrument used for data collection, specifying make and model number.*
Software	*Describe the software used to collect and analyze the flow cytometry data. For custom code that has been deposited into a community repository, provide accession details.*
Cell population abundance	*Describe the abundance of the relevant cell populations within post-sort fractions, providing details on the purity of the samples and how it was determined.*
Gating strategy	*Describe the gating strategy used for all relevant experiments, specifying the preliminary FSC/SSC gates of the starting cell population, indicating where boundaries between "positive" and "negative" staining cell populations are defined.*

- [ ] Tick this box to confirm that a figure exemplifying the gating strategy is provided in the Supplementary Information.

Magnetic resonance imaging

Experimental design

Design type	*Indicate task or resting state; event-related or block design.*

Design specifications	*Specify the number of blocks, trials or experimental units per session and/or subject, and specify the length of each trial or block (if trials are blocked) and interval between trials.*
Behavioral performance measures	*State number and/or type of variables recorded (e.g. correct button press, response time) and what statistics were used to establish that the subjects were performing the task as expected (e.g. mean, range, and/or standard deviation across subjects).*

Acquisition

Imaging type(s)	*Specify: functional, structural, diffusion, perfusion.*
Field strength	*Specify in Tesla*
Sequence & imaging parameters	*Specify the pulse sequence type (gradient echo, spin echo, etc.), imaging type (EPI, spiral, etc.), field of view, matrix size, slice thickness, orientation and TE/TR/flip angle.*
Area of acquisition	*State whether a whole brain scan was used OR define the area of acquisition, describing how the region was determined.*

Diffusion MRI ☐ Used ☐ Not used

Preprocessing

Preprocessing software	*Provide detail on software version and revision number and on specific parameters (model/functions, brain extraction, segmentation, smoothing kernel size, etc.).*
Normalization	*If data were normalized/standardized, describe the approach(es): specify linear or non-linear and define image types used for transformation OR indicate that data were not normalized and explain rationale for lack of normalization.*
Normalization template	*Describe the template used for normalization/transformation, specifying subject space or group standardized space (e.g. original Talairach, MNI305, ICBM152) OR indicate that the data were not normalized.*
Noise and artifact removal	*Describe your procedure(s) for artifact and structured noise removal, specifying motion parameters, tissue signals and physiological signals (heart rate, respiration).*
Volume censoring	*Define your software and/or method and criteria for volume censoring, and state the extent of such censoring.*

Statistical modeling & inference

Model type and settings	*Specify type (mass univariate, multivariate, RSA, predictive, etc.) and describe essential details of the model at the first and second levels (e.g. fixed, random or mixed effects; drift or auto-correlation).*
Effect(s) tested	*Define precise effect in terms of the task or stimulus conditions instead of psychological concepts and indicate whether ANOVA or factorial designs were used.*

Specify type of analysis: ☐ Whole brain ☐ ROI-based ☐ Both

Statistic type for inference (See [Eklund et al. 2016])	*Specify voxel-wise or cluster-wise and report all relevant parameters for cluster-wise methods.*
Correction	*Describe the type of correction and how it is obtained for multiple comparisons (e.g. FWE, FDR, permutation or Monte Carlo).*

Models & analysis

n/a	Involved in the study
☐	☐ Functional and/or effective connectivity
☐	☐ Graph analysis
☐	☐ Multivariate modeling or predictive analysis

Functional and/or effective connectivity	*Report the measures of dependence used and the model details (e.g. Pearson correlation, partial correlation, mutual information).*
Graph analysis	*Report the dependent variable and connectivity measure, specifying weighted graph or binarized graph, subject- or group-level, and the global and/or node summaries used (e.g. clustering coefficient, efficiency, etc.).*
Multivariate modeling and predictive analysis	*Specify independent variables, features extraction and dimension reduction, model, training and evaluation metrics.*