nature machine intelligence

ARTICLES

https://doi.org/10.1038/s42256-021-00431-x

Check for updates

Hierarchical deep reinforcement learning reveals a modular mechanism of cell movement

Zi Wang[1,4], Yichi Xu[2,4], Dali Wang[1,3] ✉, Jiawei Yang[1] and Zhirong Bao[2] ✉

Time-lapse images of cells and tissues contain rich information about dynamic cell behaviours, which reflect the underlying processes of proliferation, differentiation and morphogenesis. However, we lack computational tools for effective inference. Here we exploit deep reinforcement learning (DRL) to infer cell–cell interactions and collective cell behaviours in tissue morphogenesis from three-dimensional (3D) time-lapse images. We use hierarchical DRL (HDRL), known for multiscale learning and data efficiency, to examine cell migrations based on images with a ubiquitous nuclear label and simple rules formulated from empirical statistics of the images. When applied to *Caenorhabditis elegans* embryogenesis, HDRL reveals a multiphase, modular organization of cell movement. Imaging with additional cellular markers confirms the modular organization as a novel migration mechanism, which we term sequential rosettes. Furthermore, HDRL forms a transferable model that successfully differentiates sequential rosettes-based migration from others. Our study demonstrates a powerful approach to infer the underlying biology from time-lapse imaging without prior knowledge.

Recent applications of deep learning have demonstrated great power in image processing and image analysis in biology and biomedicine in terms of image reconstruction, classification, segmentation and augmentation[1,2]. Meanwhile, three-dimensional (3D) time-lapse images contain rich information on dynamic cell behaviours such as cell division, cell migration and collective cell behaviours, which in turn reflect the diverse processes of proliferation, differentiation and morphogenesis[3–5]. In particular, cell–cell interactions produce forces and recognizable features such as movement, shape changes and the spatial configuration of neighbouring cells[6]. These features can be explored to infer hidden cell–cell interactions and potentially identify novel biology. The supreme ability of deep learning to capture intricate relationships in features offers great opportunities in this regard. However, it is not yet clear what learning strategies and approaches would be productive.

Deep reinforcement learning (DRL), which formulates the dynamic decision-making problem with a Markov decision process (MDP), has been highly successful in solving dynamic, global optimization problems such as game-playing[7–10] and robotic manipulation[11–13], achieving or surpassing human performance. These problems involve learning to optimize a sequence of actions, where each action is based on the current state of a dynamic environment, and the sequence of actions achieves a global optimum[14]. Although the time-sequence scheme of DRL is applicable to learning dynamic cell behaviours, it typically requires millions of training data and days of computation[15], and its application in bioimage analysis has been limited[1,2].

There are a variety of DRL approaches, but hierarchical deep reinforcement learning (HDRL)[16,17] emphasizes the use of subgoals, that is, meaningful intermediate achievements. For example, in a two-level hierarchy, the higher-level module learns to choose the sequence of subgoals to achieve the overall task and the lower-level module learns the sequence of actions to achieve a subgoal[18,19]. As to the reward, the value functions combine local feedback for the action at each time step with long-range feedback for achieving subgoals and the ultimate goal. The use of subgoals reduces the search space and the demand for sample size[15], and the optimal solution can be found in an acceptable time.

In this article we exploit HDRL to learn cell behaviours during cell migration from time-lapse images, treating the migrating cell as an agent, the other cells in the images as the environment, and the migration process as a series of intermediate destinations (subgoals). Emergent attributes of the learned migratory behaviour, such as the choice of subgoals and the associated movement patterns, are collected to examine the underlying biology in terms of potential cell–cell interactions and collective cell behaviours. We further hypothesize that, after successful learning, the feature extraction component of the policy network in the lower module of HDRL, which functions to capture the salient features of the environment, provides an effective representation of the cell behaviours and cell–cell interactions, and we test this hypothesis with a novel transfer learning experiment in which the feature extraction component in the lower-level module is transferred into a new image classifier to recognize additional cases of cell migration driven by the same cellular mechanism.

Using this approach, we examine cell migrations in *Caenorhabditis elegans* embryogenesis[20]. The 3D time-lapse images contain minimal labelling and annotation of cells, and only the position and size of the cell nuclei are provided. The reward system consists of a global feedback (reaching the destination in a specified time) and local rules compiled from observational data to guide cell movement actions at each time step, with the overall goal to arrive at the destination within a specified time but not necessarily copying the observed path. Our method reveals the modular organization of cells during cell migration. Subsequent imaging and genetic perturbations confirm these modules as sequential formation of multicellular rosettes involving the migrating cell and its neighbours, which is a novel mechanism of cell migration that we term sequential rosettes. A new classifier created through transfer learning successfully identifies additional cases of cell migration driven

[1]Department of Electrical Engineering and Computer Science, University of Tennessee, Knoxville, TN, USA. [2]Developmental Biology Program, Sloan Kettering Institute, New York, NY, USA. [3]Environmental Science Division, Oak Ridge National Laboratory, Oak Ridge, TN, USA. [4]These authors contributed equally: Zi Wang, Yichi Xu. ✉e-mail: wangd@ornl.gov; baoz@mskcc.org

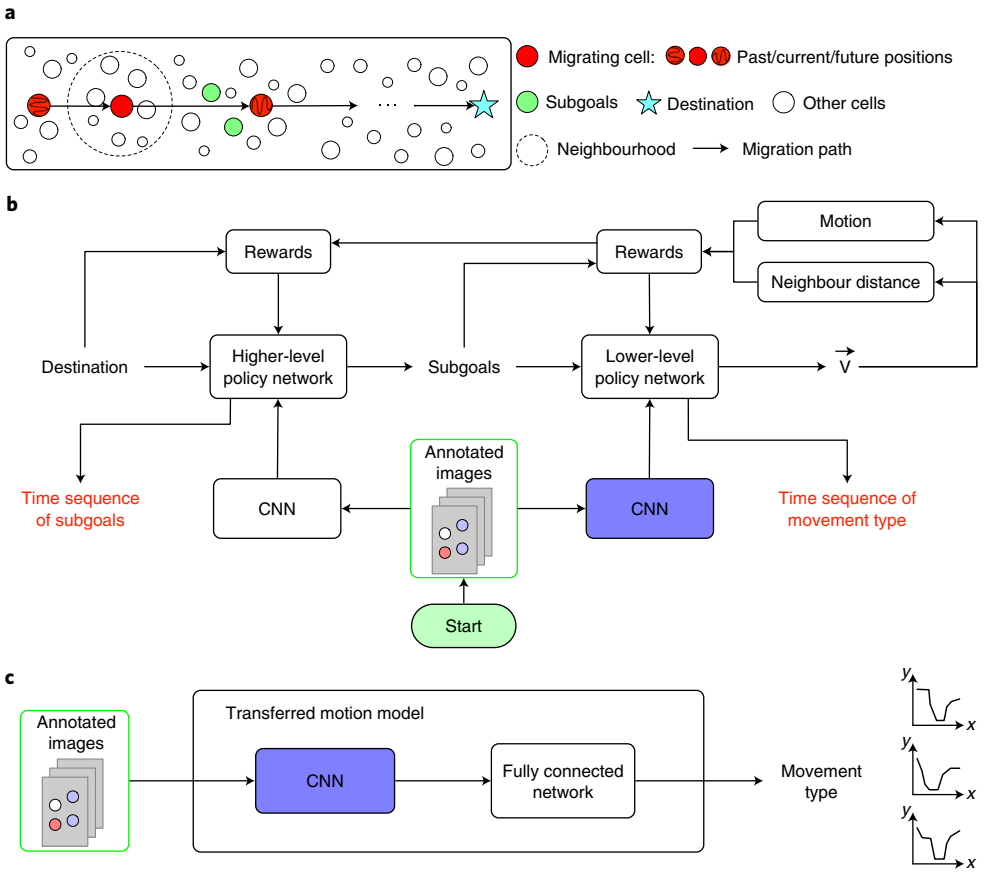

Fig. 1 | Concepts and design to model cell movement with HDRL. a, A schematic showing the actors in cell movement modelling. Small circles represent cell nuclei in a tissue. The migrating cell, subgoals and other cells are coloured red, green and white, respectively. The dashed circle indicates the current neighbourhood of the migrating cell. The migrating cell moves along the arrows towards the destination marked with a cyan star. **b**, Architecture and major components of the two-level HDRL used. Arrows indicate data flow between components. The green-bounded box indicates the input of the CNNs (feature extraction component). Black-bounded boxes indicate model components. Model inputs and outputs are shown as text/symbols without bounding boxes (red indicates output). White and blue CNNs indicate separate networks with their own parameters. **c**, Architecture of the TMM. The blue CNN indicates that it is transferred from the CNN of the lower-level module in the HDRL framework in **b**.

by sequential rosettes and distinguishes those that are not, suggesting effective representation of cell behaviours in HDRL. Our study demonstrates that HDRL can be used to form informative models of dynamic cell behaviours with simple rules and a small observational dataset, and to discover emergent features of cells and tissues without prior knowledge.

Result

Design and scheme. *Image data and set-up for reinforcement learning.* The studied images were 3D time-lapse images in which cells were labelled with a ubiquitously expressed nuclear marker[21]. The nuclei were segmented and tracked to obtain information about nuclear sizes and positions over time[22–24].

We used the 3D time-lapse images to create a reinforcement learning system (Methods). Specifically, we treated the migrating cell of interest as a learning agent and the other cells in the images as the environment. To construct this environment, the positions of the environmental cells over time were cloned from the input image data. To further enhance the dynamic scenes of the environment, we increased the temporal resolution by tenfold, interpolating cell positions with a small injection of randomness (Methods). We then annotated elements in the environment that were key for learning, namely a migration destination (a designated cell in the environment), subgoals (cells selected from the environment) and neighbour relationship among cells at every time point (Fig. 1a).

For the migrating cell, its starting position was copied from the input image data, and the subsequent positions over time were generated by the sequence of movement actions of the agent. For simplicity, the migrating cell moved at a constant speed, which was estimated as the average speed in observational images. The agent learned to choose the direction of movement at each time point.

Reward construction. The rewards in our system consist of long-range and local feedback on cell movement (Fig. 1b). The long-range feedback includes a global reward for reaching the specified destination, as well as rewards for achieving each subgoal. The local feedback consists of rules that are used to score cell movement at each time step.

To develop the local rules, we assume that a period of directional movement of the migrating cell is an indication of a directional net force acting on the cell, and that the spatial configuration of cells around the migrating cell reflects the forces. Two models were developed to serve as the local rules.

The first model is termed the motion model and comprises a convolutional neural network (CNN) that calls directional versus random movement at a given time point. The CNN was trained on observational images that were labelled directional or random after considering the correlation of the velocity vectors in a period of time. To focus the learning on features of the local neighbourhood, images were cropped to include only the migrating cell and its direct neighbours.

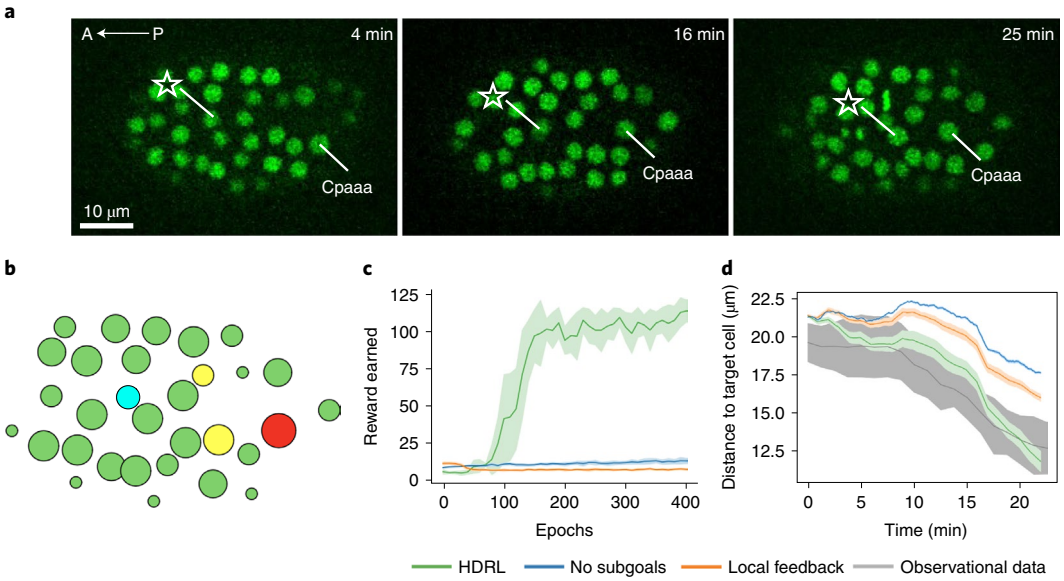

Fig. 2 | Modelling Cpaaa migration in *C. elegans* embryogenesis. a, Micrographs of a *C. elegans* embryo showing the migration of Cpaaa (dorsal view, anterior to the left; A, anterior; P, posterior). Nuclei are shown in green. The star indicates the ABarpaapp cell, which is contacted by Cpaaa when Cpaaa reaches its destination. Time 0 is the birth of Cpaaa. **b**, Annotation of the image at a time point during HDRL training (Supplementary Video 2). Red, yellow, cyan and green indicate the migrating cell, subgoals, the destination and other cells in the embryo, respectively. **c**, The rewards generated over training epochs with different rule settings. 'HDRL' indicates HDRL training with global feedback (for reaching the destination), subgoals and local feedback (from the motion model and neighbour distance model). 'No subgoals' indicates DRL training with global feedback and the local feedback but no subgoals. 'Local feedback' indicates training with only the local feedback. The earned rewards were averaged based on five runs and the shaded regions indicate 1 s.d. **d**, Position of the migrating cell (distance to the target cell) over time after training with different rule settings. 'Observational data' indicates Cpaaa migration in the image series. Lines represent the average of each group (20 runs for each rule setting and 10 wild-type embryos for observational data) and shaded regions indicate 1 s.d. Time 0 is 15 min after the birth of Cpaaa.

The second model is termed the neighbour distance model and considers the acceptable minimal distance between neighbouring cells. The distribution of such distances was compiled from observational data.

HDRL for model formation. An HDRL with a two-level architecture was used to learn how to guide the migrating cell to reach a given destination in time (Fig. 1b and Supplementary Table 1). The lower- and higher-level RL modules each contains a policy network, which contains a CNN that serves as the feature extraction component to learn features of the input image series, and a fully connected network that makes its actions on top of the CNN. Images of the entire embryo were provided as input, allowing the system to learn features beyond the immediate vicinity of the migrating cell. The policy network in the lower-level module uses the image features projected from its CNN to generate a suitable movement at each time point (direction of movement with a quantal step size) to reach a given subgoal, and the higher-level policy network uses the image features from its own CNN to select a sequence of subgoals for the lower-level module.

We used the hierarchical Deep Q-Network (h-DQN)[18], which allows for flexible subgoal specification, to integrate value functions based on the local and long-range feedback. Subgoals were chosen from the annotated nuclei that are secondary neighbours of the migrating cell (neighbour of a neighbour). For fast and stable learning, we chose a pair of secondary neighbours at a time as a potential subgoal. A subgoal is considered achieved if both cells become stable neighbours of the migrating cells that last over a certain time period. Similarly, the final destination is considered achieved if the designated destination cell becomes a stable neighbour of the migrating cell (Fig. 1a). Neighbour relationship among cells is determined by a neighbour relationship model in real time with a fast random forest classifier that was trained to approximate Voronoi neighbours (Methods).

To gain more insights into the migration process, we collected all successful time sequences of subgoals (Fig. 1b), as opposed to the typical single optimal sequence. For each successful sequence, we also recorded the movement types (directional versus random) at each time step (Fig. 1b).

Deployment of the learned model through transfer learning. We hypothesize that the CNN in the lower-level HDRL module constitutes an effective model that represents the examined cell movement. To test this hypothesis and to exploit the potential model for novel biology, we devised a transfer learning approach to create an image-based classifier of cell movement types. Specifically, the CNN in the lower-level HDRL module was fused to a fully connected network (Fig. 1c). This classifier, which we term the transferred motion model (TMM), was trained on labelled observational images of the same cell migration process analysed by HDRL, with the CNN being fixed to serve the purpose of testing the HDRL-trained CNN. The TMM was then tested on images of other cell migration processes to determine whether or not they use the same underlying mechanism as in the HDRL training case (and therefore share key cellular and tissue features).

HDRL model formation in *C. elegans* embryogenesis. We applied our method to examine a cell migration event during *C. elegans* embryogenesis. A cell named Cpaaa is born at the dorsal posterior side of the embryo and migrates anteriorly around 15 min after its birth[25]. Cpaaa intercalates in between two rows of cells of the ABarp lineage. The migration ends when Cpaaa becomes the neighbour of the ABarpaapp cell in ~25 min (Fig. 2a and Supplementary Video 1). Cpaaa is annotated as the migrating cell and ABarpaapp the destination.

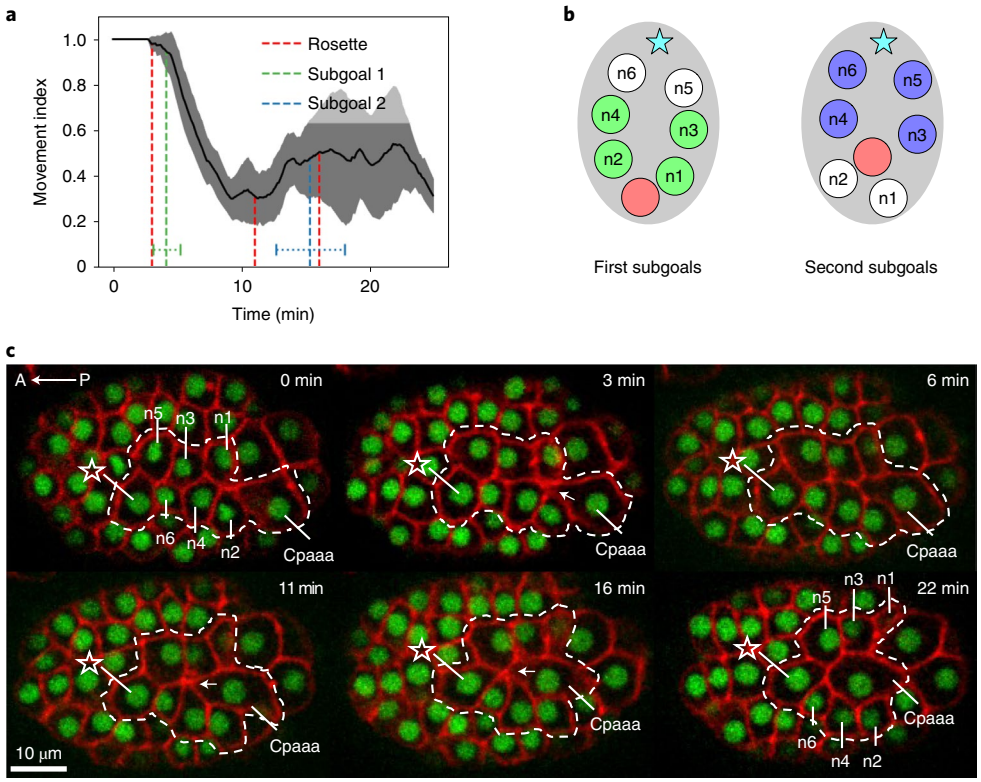

Fig. 3 | Modular organization of Cpaaa movement in HDRL and 3D time-lapse imaging. a, The MI curve of successful migration scenarios. The average MI (black line) and 1 s.d. (shaded region) are shown. Green and blue dashed lines indicate the timing of achieving the first and second subgoal, respectively, with the vertical lines indicating the average time and the horizontal lines 1 s.d. Red lines indicate the timing of rosette formation identified from the imaging experiments shown in **c**. Time 0 is 15 min after the birth of Cpaaa. **b**, The collection of all the subgoals in each of the directional movement phases. The group of subgoals for the first phase includes n1–n4 (green) and the group for the second phase n3–n6 (blue). Red circle, Cpaaa; cyan star, ABarpaapp. **c**, Micrographs of the *C. elegans* embryo showing sequential rosettes during Cpaaa migration (dorsal view, anterior to the left; A, anterior; P, posterior). Nuclei are shown in green and cell membranes in red. Time 0 is 15 min after the birth of Cpaaa. Dashed lines encircle the eight cells involved. Arrows indicate the centres of three rosettes. Star, ABarpaapp; n1, ABarppapp; n2, ABarppppa; n3, ABarppapa; n4, ABarppppap; n5, ABarppaap; n6, ABarppppaa.

As the input, embryos were imaged at 1-min interval, and nuclei were segmented and tracked as described in ref. [26]. The motion model and neighbour distance model were trained on 50 wild-type embryos[27]. For HDRL, the time step was set to 6 s, a tenfold upsampling from the observational data. The environment was based on the image series of a wild-type embryo. A typical successful scenario lasted for around 250–300 time steps. Through the simulation, HDRL thoroughly examined ~120 scenarios (sequences of potential subgoals) and identified a total of 21 successful scenarios.

When both the local and longer-term feedback were used, the migrating cell earned meaningful rewards that guided cell movement through the learning process (Fig. 2b,c and Supplementary Video 2). At the end of learning, the migration path of Cpaaa approximated the path in the observational data (Fig. 2d). By contrast, when not using subgoals, the reward did not increase over epochs and, at the end of learning, Cpaaa failed to move towards the destination, demonstrating the power of using subgoals in HDRL. Similarly, with local feedback alone, the system also failed to learn. Although not surprising, the simple, common-sense local rules compiled from observational data alone are not sufficient to correctly direct cell movement, but combination with the global feedback of the destination allows successful learning of cell movement.

HDRL reveals modular organization of Cpaaa migration. To better understand the successful scenarios of Cpaaa migration, we examined the temporal pattern of directional movement and the set of subgoals. In terms of the temporal pattern of directional movement, our HDRL reports the Cpaaa movement type (directional as '1' and random as '0') at each time step. We smoothed the sequence of movement types over a sliding window of 30 time steps to calculate a value for the movement index (MI), which better illustrates sustained directional movement (Supplementary Fig. 1). The mean MI of the 21 successful scenarios (Fig. 3a) revealed two phases of directional movement, from minute 0 to 5 and from minute 15 to 22 after Cpaaa started to migrate.

The set of subgoals in the 21 successful scenarios also suggests a stepwise organization of movement. Each of the 21 scenarios involved a sequence of two subgoals. All subgoals consisted of the two rows of ABarp cells that Cpaaa migrates through in observational data. For simplicity, these ABarp cells are denoted n1–n6 (Fig. 3b). Among the 21 scenarios, the first subgoals involved n1–n4 (green, Fig. 3b) and the second subgoals n3–n6 (blue, Fig. 3b). The two subgoals were achieved, on average, around minutes 4.1 and 15.3, respectively, which correspond to the two phases of directional movement (green and blue lines, Fig. 3a).

Collective behaviour explains organization of cell movement. To better understand the modular organization of the Cpaaa movement, we performed two-colour 3D time-lapse imaging of embryogenesis with a broadly expressed cell membrane marker to assess cell shape dynamics on top of the ubiquitously expressed histone marker for cell tracking[26].

The images revealed a striking collective behaviour of Cpaaa and its neighbouring cells to mediate its migrations (Fig. 3c). Specifically,

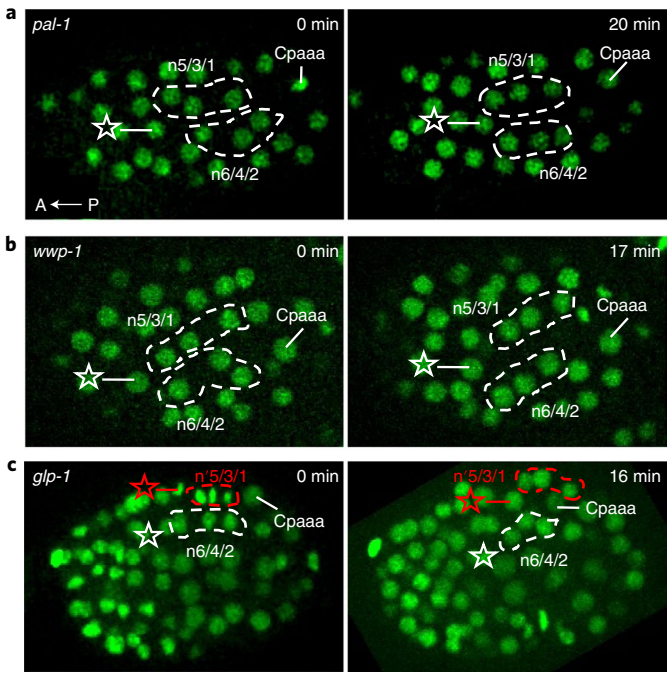

Fig. 4 | Migration of Cpaaa upon genetic perturbations. Micrographs of the *C. elegans* embryo. See Fig. 2a for the convention. The white star marks the destination in the wild type (ABarpaapp). White dashed circles mark the ABarp cells involved. For n1–n6, see Fig. 3. Red star, dashed circles and n′ mark cells in the ABprp lineage that have adopted the corresponding ABarp fates. Time 0 is 15 min after the birth of Cpaaa. **a**, A *pal-1*(*RNAi*) embryo. **b**, A *wwp-1*(*RNAi*) embryo. **c**, A *glp-1*(*RNAi*) embryo. Red star, ABprpaapp; n′1, ABprppapp; n′3, ABprppapa; n′5, ABprppaap. Note that the wild-type destination (ABarpaapp) is on the dorsal side of the plane shown and the white star indicates the projection of the ABarpaapp position on the plane shown. n1/3/5 in the ABarp lineage are also on the dorsal side and not shown.

Cpaaa forms a sequence of multicellular rosettes with the ABarp cells during its migration (dashed regions, Fig. 3c). Multicellular rosettes are recognized by morphology where five or more cells converge to a point[28]. Three rosettes form over time with sequential edge contraction and resolution events (arrows, Fig. 3c). The formation and resolution of each rosette are correlated with Cpaaa movement anteriorly towards the ABarpaapp cell (stars in Fig. 3c), so that when the last rosette resolves, Cpaaa forms contact with ABarpaapp and stops migrating. The sequence of rosettes is stereotypical among embryos ($n=6$ embryos) in terms of the cell composition and timing of each rosette. The first one forms at approximately minute 3 after Cpaaa starts to migrate, which involves Cpaaa and four ABarp cells (n1–n4) and resolves around minute 6. The second rosette forms at approximately minute 11, involving Cpaaa and n3–n6. The third rosette forms at approximately minute 16, involving Cpaaa, n4–n6 and the target cell ABarpaapp, and resolves by minute 22.

Although the formation and resolution of multicellular rosettes are known to mediate cell intercalation[28], the sequence of rosettes formed by Cpaaa and the ABarp cells is distinct from known cases. The known cases of rosettes occur in isolation in that a rosette forms and resolves. By contrast, the sequence of rosettes here involves overlapping populations of cells and a temporally coordinated order of edge contraction and resolution, which indicates an additional level of coordination mechanism. We therefore term this collective behaviour 'sequential rosettes'.

The sequence of three rosettes naturally delineates the Cpaaa movement into three steps that could explain the modular organization revealed by HDRL. The first rosette corresponds to the first phase of directional movement revealed by HDRL in terms of timing (minute 0 to minute 6 versus minute 0 to minute 5; Fig. 3a), and the set of subgoals (n1–n4) corresponds to the cell composition of the first rosette (Cpaaa + n1–n4). The second and third rosettes largely overlap in cell composition in terms of the ABarp cells (n3–n6 versus n4–n6), which correspond to the second subgoals (n3–n6). The timing of the second phase of directional movement in the averaged MI curve (minutes 15 to 22) overlaps with the end of the second rosette (minutes 11 to 15) and the third rosette (minutes 16 to 22). The average MI in the second phase shows increased variance, indicating heterogeneity of the paths among individual scenarios. Indeed, the MI curve of individual scenarios (Supplementary Fig. 1) shows peaks, the timings of which correspond to the second or the third rosette in a subset of scenarios.

Local cell–cell interactions underlie sequential rosettes. Rosette formation involves a group of neighbouring cells. The stereotypical features of the sequential rosettes in terms of cell composition and order of rosettes further indicate a hypothesis where specific cell–cell interactions among appropriate cell types underlie the dynamic behaviours. To test this hypothesis, we examined situations where the fates and types of the participating cells were perturbed. We analysed two sets of perturbations where RNA interference (RNAi) or knockdown of genes changed the fate in different cells[29].

In the first set, the fates of Cpaaa and/or the ABarp cells were perturbed. Loss of function or RNAi of the *pal-1*/Caudal gene, which encodes a conserved transcription factor that activates the expression of other genes, abolishes the C fate but does not appear to affect other lineages[29,30]. In *pal-1*(*RNAi*) we found that Cpaaa failed to intercalate into the ABarp cells and stayed where it was born (Fig. 4a, six out of six embryos). This phenotype suggests that the C fate is required and rules out the possibility that the ABarp cells are capable of pulling in any neighbouring cells. The *wwp-1* gene encodes a conserved E3 ubiquitin ligase that marks specific protein targets for degradation. In the early *C. elegans* embryo, *wwp-1*(*RNAi*) causes ectopic Notch signalling in the ABarp cell so that the ABarp lineage adopts the fate of the ABprp lineage and produces neurons instead of skin cells, but does not appear to affect the C lineage[29]. In *wwp-1*(*RNAi*), Cpaaa also failed to intercalate into the ABarp cells (Fig. 4b, seven out of seven embryos). This phenotype suggests that the ABarp fates are required and rules out the possibility that the Cpaaa cell is capable of engaging any neighbouring cells for migration. We conclude that the Cpaaa movement requires cell–cell recognition and specific interaction between Cpaaa and the ABarp fates.

In the second set, the Cpaaa and ABarp cells were not affected, but an ectopic set of cells with the ABarp fates were produced next to Cpaaa. This situation was achieved by RNAi of the *glp-1*/Notch gene, which encodes the receptor in the Notch signalling pathway. In *glp-1*(*RNAi*) the ABprp cells adopted the ABarp fates (red dashed circle, Fig. 4c, five out of seven embryos)[29]. In two out of the five embryos where ABprp cells adopted the ABarp fates, Cpaaa migrated normally and intercalated into the ABarp cells. In the remaining three embryos, Cpaaa intercalated between the ABarp and ABprp cells. Interestingly, the ABprp cells involved are the corresponding cells as in the wild-type ABarp cells in terms of lineage identity (Fig. 4 legend).

Taken together, these perturbations support the notion that local interactions among specific cells give rise to the emergent collective cell behaviours of rosette formation and cell migration.

Transfer learning from HDRL distinguishes movement patterns. The above results establish sequential rosettes as a novel mechanism that underlies the migration of Cpaaa. Intriguingly, although HDRL was not aware of this mechanism, the emergent features from HDRL,

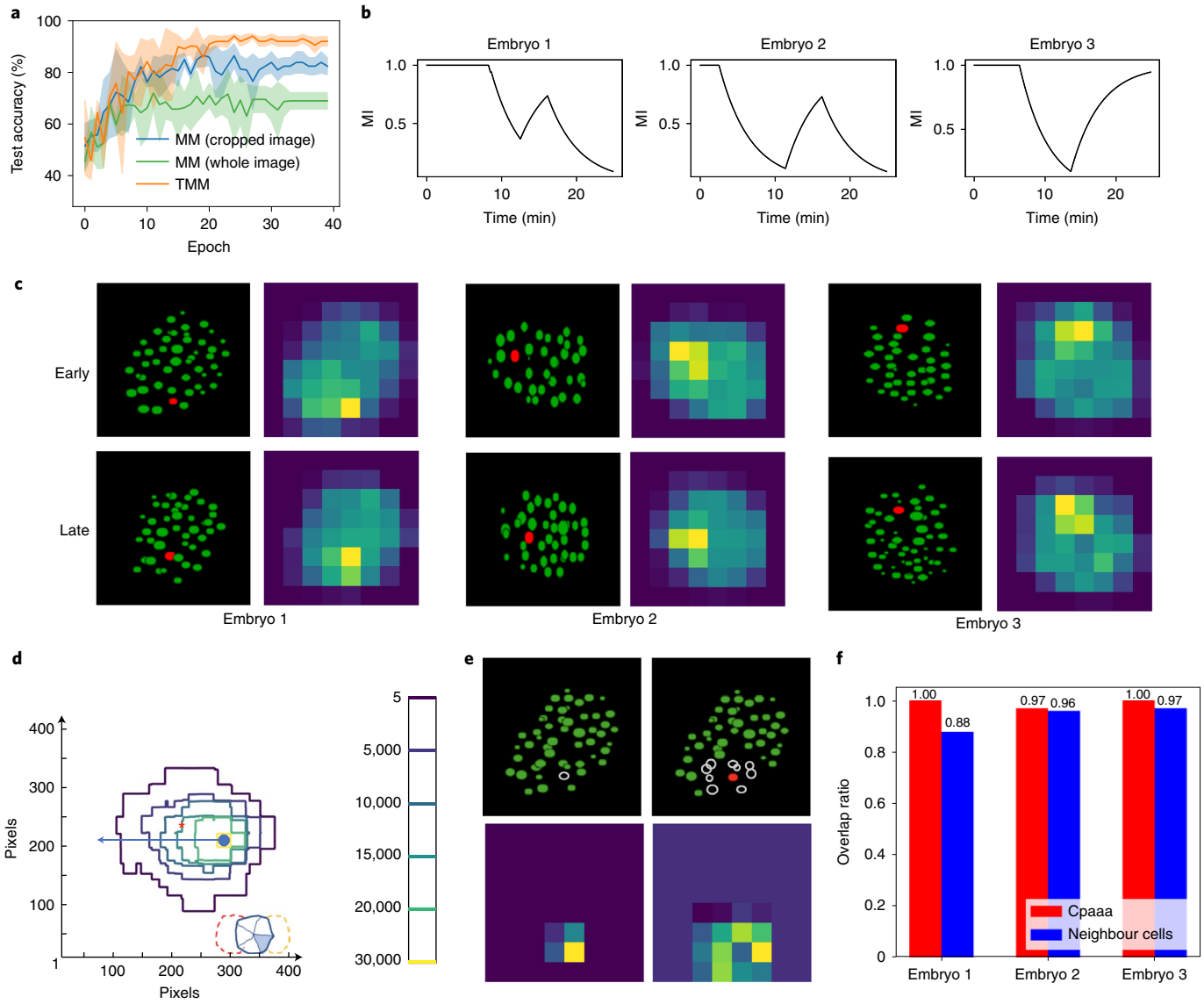

Fig. 5 | Validation and characterization of the TMM. a, Test accuracy of the TMM and the motion model (MM). Each line is the average of five runs. The shaded regions indicate 1 s.d. **b**, The MI curve from the TMM in three embryos that were not included in the training set. Time 0 is 15 min after the birth of Cpaaa. **c**, Example input images and the corresponding summary feature maps of the TMM at an early and a late time point of Cpaaa migration in three embryos. Early and late time points are around 18 and 42 min after the birth of Cpaaa, respectively. Input images are shown on the left and feature maps on the right. In the images, the Cpaaa nucleus is in red and other cells in green. In the feature maps, colours represent the value at each pixel of the summary map. Warmer colours represent higher values. **d**, Isocontour representation of the aggregated value of feature maps across embryos and time points. The filled blue circle marks the position of the migrating cell and the blue arrow marks the migration direction. Colours of the isocontour represent the aggregated values, with warmer colours representing higher values. The red asterisk marks the isocontour that encircles a pixel area of approximately 150×100. The bottom inset shows a schematic of the migrating cell (shaded blue), current rosette neighbours (white cells in the blue circle) as well as the other neighbours of the previous (yellow) and the next (red) rosette. **e**, Upper panels: example ablated input images from embryo 1 after ablating Cpaaa (left) or Cpaaa's neighbour cells (right). White open circles mark the positions of the ablated cells. Lower panels: the corresponding summary difference maps, which show changes in the summary feature maps after ablating Cpaaa (left) and Cpaaa's neighbour cells (right). Warmer colours represent higher values. **f**, The ratio of overlapped area between the effective areas and the ablated cells (Cpaaa (red) and Cpaaa's neighbour cells (blue)) to the effective area among a total of 74 time steps in three embryos. Supplementary Fig. 4 provides the underlying data.

namely the set of subgoals and phases of directional movement, closely reflect the collective cell behaviour of rosette formation. In this section, we further examine to what extent the HDRL-trained CNN provides a model for rosette-based cell migration.

As outlined in Fig. 1c, the CNN of the lower-level HDRL module was connected to a fully connected neural network to create the TMM and classify whether a cell movement is based on sequential rosettes or not. The TMM was first trained to classify the movement type of Cpaaa, with the transferred CNN being fixed. The training data were the same set of embryos used to train the motion model. The TMM was trained and tested on images of the whole embryo, because the transferred CNN was trained on whole embryo images during HDRL model formation. By contrast, the motion model was trained on cropped images that contained just the local neighbourhood of the migrating cell. For comparison, it was tested on both cropped and whole images.

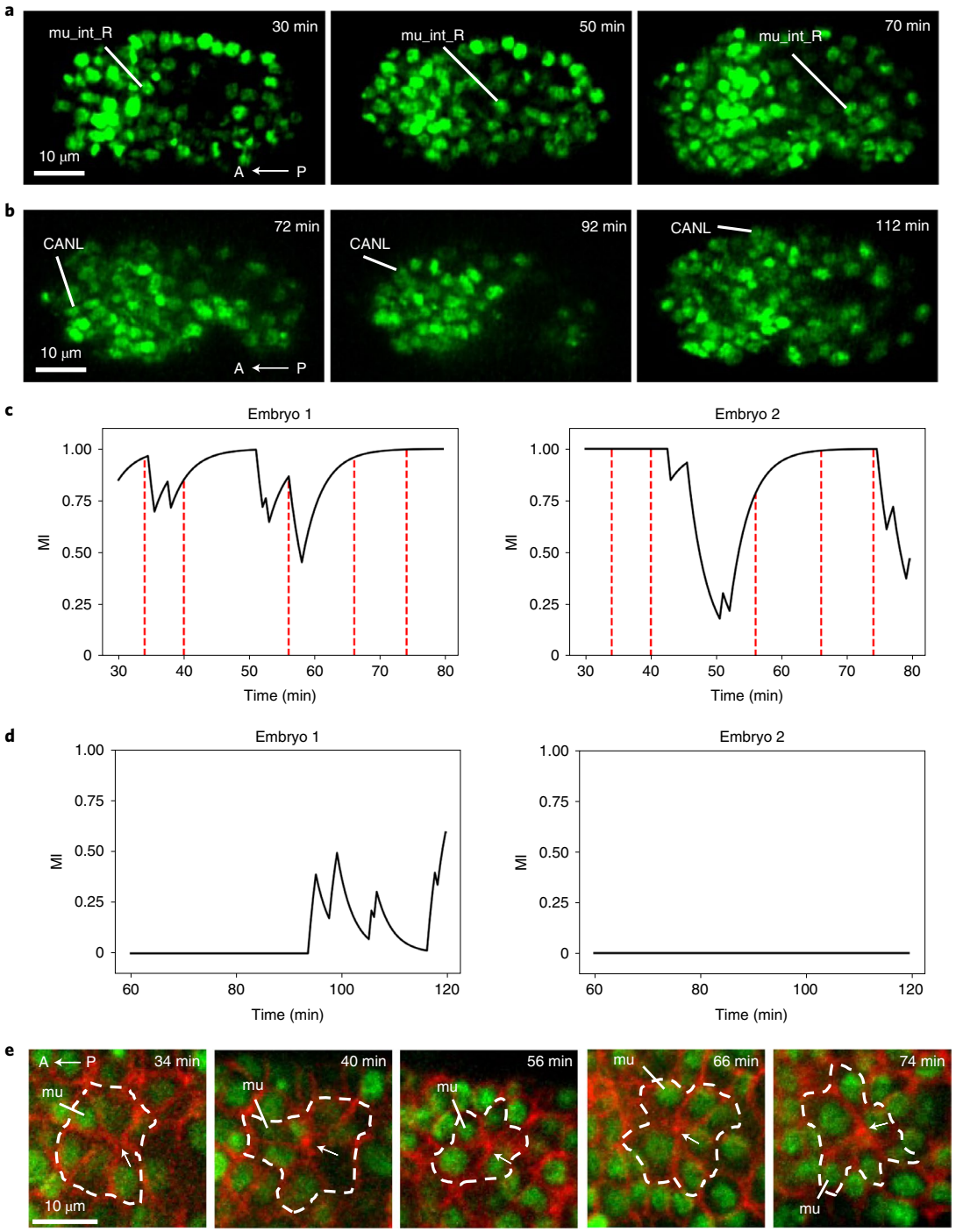

Fig. 6 | TMM classification and 3D time-lapse imaging of mu_int_R and CANL migration. a,**b**, Micrographs of a *C. elegans* embryo showing the migration of mu_int_R (**a**) and CANL (**b**). See Fig. 2a for the convention. Time 0 is the birth of mu_int_R and CANL, respectively. **c**,**d**, The MI curve from the TMM over mu_int_R (**c**) and CANL (**d**) migration in two embryos. Time 0 is the birth of mu_int_R and CANL, respectively. Red dashed lines indicate the timings of rosette formation identified from the imaging experiments shown in **e**. **e**, Micrographs of the *C. elegans* embryo showing sequential rosettes during mu_int_R (marked by 'mu') migration (dorsal view, anterior to the left). Nuclei are shown in green and cell membranes in red. The dashed lines show the contours of the rosettes. The arrows indicate the rosette centres. Time 0 is the birth of mu_int_R.

Directional movement of Cpaaa was labelled as rosette-based movement and random movement of Cpaaa as non-rosette-based movement.

The TMM outperformed the motion model with a clear margin when classifying Cpaaa movement (Fig. 5a). With the same training/testing split (Methods), the average test accuracy of the TMM reached over 90%. The accuracy of the motion model when using cropped images averaged over 80%. When whole embryo images were used, the accuracy of the motion model dropped to below 70%. These results demonstrate the effectiveness of HDRL-based training of the CNN.

We further tested the TMM on Cpaaa movement using three embryos that were not in the training set. Without any further fine-tuning, the TMM successfully revealed the phases of

rosette-based movement in the MI plots for each of the three embryos (Fig. 5b).

We then asked how the HDRL-trained CNN may have captured the features of rosette-based cell migration. To this end, we examined the feature maps generated for the three embryos used as the test set in Fig. 5b. At each time point, the CNN generated 64 feature maps with dimensions of 8×8 (Supplementary Fig. 2). We also summed the 64 maps as a summary for the time point (Fig. 5c). These summary maps as well as many of the individual maps appear to highlight the shape and size of the embryo, the position of the migrating cell, as well as the neighbouring cells in the direction of migration, regardless of the overall orientation of the embryo. For a more quantitative analysis of these observations we aligned the summary maps (of representative time points at the early, middle and late stages of migration in each embryo) by the position of the migrating cell and the migration direction, and upsampled the 8×8 summary maps to the original resolution of the input microscopic images. We further aggregated the upsampled maps of the three embryos and created an isocontour plot to summarize the activation levels relative to the migrating cell (Fig. 5d). Notably, an isocontour encircles ~150×100 pixels around the migrating cell (red asterisk, Fig. 5d), which is spatially correlated to an area containing the migrating cell and its rosette neighbours, as well as the neighbours participating in the previous and the next rosette, respectively (bottom inset, Fig. 5d). Thus, the highest level of activation among the feature maps is correlated to the position of the migrating cell and its neighbours.

To further test whether the CNN responds to the migrating cell and its neighbours, we conducted two ablation experiments. Specifically, we removed the migrating cell (Cpaaa) or its neighbour cells from the input images and examined the changes in the feature maps. Each embryo at each time point produced 64 feature maps and 64 difference maps between the corresponding feature maps from the original and the ablated input images (Supplementary Fig. 3). The 64 difference maps at each time step were summed into a summary difference map for further analysis (Fig. 5e). We focused on the pixels in a summary difference map with clear differences (Methods), which we refer to as the effective area, and asked how well the effective areas overlap with the ablated cells spatially. Across a total of 74 time steps in three embryos, 97–100% of the pixels in the effective areas overlap with Cpaaa when Cpaaa is ablated and 88–97% of pixels in the effective area overlap with the neighbour cells when neighbour cells are ablated (Fig. 5f and Supplementary Fig. 4). These results show that the CNN responds to the migrating cell and its neighbours and that these cells contribute largely locally to the feature maps in their corresponding spatial area.

Finally, to test whether the HDRL-trained CNN provides an effective model for rosette-based cell migration, we applied the TMM, which was trained on the Cpaaa cell, to classify other long-range cell migrations in *C. elegans* embryogenesis. Based on the documentation of long-range migrations in the literature[20], we focused on two cells with the largest migration distance, namely mu_int_R and CANL. Both of these migrations occur ~3.5 h later than Cpaaa (500-cell stage versus 150-cell stage; Fig. 6a,b and Supplementary Videos 3 and 4), with a smaller average cell size and a higher cell density. The global direction of these migrations is from the anterior to the posterior, opposite that of Cpaaa. As such, these migrations present embryonic and cellular characteristics that are different from those of Cpaaa.

The MI curves generated by the TMM showed different modes of movement for these two cells. For mu_int_R (Fig. 6c, $n=2$ embryos), the MI curve predicted multiple phases of rosette-based movement. By contrast, the movements of CANL were not recognized as rosette-based (Fig. 6d, $n=2$ embryos), indicating underlying features or mechanisms different from those of Cpaaa.

Subsequent imaging with a cell membrane marker (Fig. 6e) revealed that mu_int_R migration is indeed mediated by sequential rosettes, while CANL is not. Specifically, for mu_int_R we found a sequence of five rosettes during its migration, occurring at approximately 34, 40, 56, 66 and 74 min after the birth of mu_int_R. The timing of these rosettes corresponds to the phases of high MI (red dashed lines, Fig. 6c). On the other hand, we did not find rosettes during CANL migration. Previous studies imaging CANL migration suggested that CANL migrates by generating lamellipodia at the leading edge, as in the canonical mechanism of cell migration[31]. These results, in which the TMM is capable of distinguishing migration driven by sequential rosettes from that driven by other mechanisms, suggest that the HDRL-trained CNN provides an effective model for rosette-based cell migration.

Discussion

Our study presents a data-driven approach to use deep learning to uncover novel biology from images. Essentially, we have shown that DRL can be used to form models of unknown cellular behaviours and inspire experimental investigations. Ultimately, our study has revealed a previously unknown mechanism of cell migration, which we term sequential rosettes.

For the DRL to form a model of dynamic cell behaviour without prior knowledge, we used CNNs as the feature extraction component of the policy networks to examine the images of the environment of the migrating cell. As demonstrated, after learning, the CNN in the lower-level module successfully represented the underlying collective cell behaviour. In additional cases of cell migration that it had not seen, the model was able to successfully distinguish sequential rosette-based migration from other types. Although it is technically difficult to explain how the neural network encodes the cell behaviours, high activation in the feature maps appears to be correlated with past, current and future neighbouring cells that form rosettes with the migrating cell. Furthermore, as shown in the test cases, this model seems insensitive to the orientation and scale of rosettes relative to the images, despite limited training data, which speaks to the advantages of CNNs for image representation.

By focusing on model formation, our study emphasizes a different perspective to DRL. Conventionally, DRL is used as a generative model to perform new tasks of the same or related nature, be it game-playing or robotic manipulation. However, in biological experiments including imaging, the typical challenge is post hoc interpretation of the observations, where the first question is whether an observation can be readily explained by known mechanisms. In such a setting, classifiers are in demand. In this regard we have shown that, after DRL training, the feature extraction component in the policy network can be directly transferred to create a classifier. This approach essentially treats DRL as a reward-guided, unsupervised training platform for model formation, and we have demonstrated the use of the feature extraction component of the policy network of DRL for transfer learning to perform other tasks.

Our study has also demonstrated HDRL as a powerful form of DRL for biology. HDRL shows a superior capability for learning and model formation with a small training set, minimal labelling and simple common-sense rules and constraints. In theory, with large amounts of training data and practically unlimited computing power for simulation, DRL is capable of learning complex processes without the greedy approach in HDRL to reduce the search space. However, such results are often not achievable. For example, in the conceptually similar problem of maze navigation, the model to represent the current position needed to be trained separately with supervised learning to achieve a multiscale representation of space and successful navigation[32]. Alternatively, a greedy reward imitating a molecular gradient of guidance has been used to achieve long-range cell migration[25]. By contrast, in this study, HDRL achieved unsupervised model formation without a strong

assumption of the underlying biological mechanism, which meets the typical situation of biological data analysis, that is, with partial knowledge and the potential for novel biology.

HDRL may thus be used broadly to study different dynamic biological processes and serve different biological questions. For example, with appropriate markers and reporters for imaging, one could examine proliferation patterns, gene expression dynamics or neuronal activities. However, each problem would require one to carefully define the space and dimensions of the dynamic process (for example, reporter level on top of (x, y, z, t)) and select biologically meaningful subgoals. More broadly, modelling a dynamic biological process as a sequence of actions in a multi-dimensional space is not limited to image-based data. However, it may not be obvious what the right type of subgoals is for every question. Furthermore, in terms of data, while HDRL reduces the amount of data needed, it may still not be trivial to label the amount of data required. For more complex questions it may still not be practical to obtain enough data experimentally. Nevertheless, as demonstrated in our study, HDRL offers an intriguing approach to exploit deep learning for cell behaviours and dynamic biological processes.

Methods

Observational dataset and annotation. The observational data in our modelling system are 3D time-lapse images in which cells are labelled with a ubiquitous nuclear marker. The size and location of each nucleus over time were extracted based on segmentation and tracking of the nuclei[22–24]. For RL, we used binary images to present the information, with each nucleus represented by a sphere of specified position and size (circular discs on different z planes). The binary images were further annotated with different colours to represent key information: a migrating cell of interest (red), a migration destination (cyan), cells selected as the subgoal (yellow) and all other cells (green). A stack of five planes centred on the migrating cell was used as input image to the different components in our system.

Reinforcement learning system set-up. The time interval in RL was set at one-tenth that of the observational data. The locations of the environmental cells were derived from observational data with a tenfold upsampling of temporal resolution and linear interpolation of cell positions. For each interpolated position, a small randomness n_l was ingested based on a normal distribution, $n_l \sim \mathcal{N}(0, 0.5)$, with the average value and standard deviation set to 0 and 0.5 pixels. The migrating cell (RL agent) was designed to move at an average speed v (obtained from the observational data) in one of the possible directions in a discrete 3D space. To enhance the robustness of the RL agent movement decisions, a small randomness n_v was ingested to the speed based on a normal distribution with standard deviation set to 10% of the average speed, $n_v \sim \mathcal{N}(0, 0.1v)$.

Neighbour relationship model. The neighbour relationship model $f_n = \{0, 1\}$ determines whether two cells are neighbours of each other based on the Voronoi diagram using the centre of each nucleus. The Voronoi neighbour relationship is approximated based on a set of criteria as described in previous work[33]. If we denote c_a and c_b as the feature vectors of cells a and b, then $f_n(c_a, c_b) = 1$ if a and b are neighbours, otherwise 0. A random forest classifier was trained as the neighbour relationship model based on over 940,000 true Voronoi neighbours/non-neighbours during *C. elegans* embryogenesis. This model achieved real-time classification during HDRL, processing a pair of cells in ~0.0002 s with 99.6% accuracy.

Motion model. The motion model was implemented to classify the movement type (directional versus random) of the migrating cell at a given moment.

Dataset. The dataset to train the motion model was established by manually labelling 50 wild-type *C. elegans* embryos. A time window covering the lifespan of the Cpaaa cell in each embryo, ~25 time points per embryo, was manually labelled by the authors, using AceTree to observe the movement pattern of the migrating cell. The correlation between the movement direction of the migrating cell in five time steps was assessed. A time point was labelled as 1 for directional movement and 0 for random movement. As the input of the neural network, the labelled images x, centred at the migrating cell, were cropped to include its neighbours (with a size of $x \in \mathbb{R}^{128\times128}$). In total, 85% of the labelled data were used as the training set and the rest as the test set.

Neural network architecture. An AlexNet style CNN f_m was used as the classifier; this consists of five convolutional layers followed by a ReLU activation layer after each. A maxpooling layer was implemented for the purpose of downsampling after the first, second and fifth convolutional layers. The convolutional layers were followed by three fully connected layers and the output of the neural network was a binary call on the two movement types, $f_m(x) \in \{0, 1\}$.

Training strategy. The neural network converged (training loss and accuracy) around 40 epochs, with an Adam optimizer, a mini-batch size of 10 and a learning rate of 0.003.

Neighbour distance model. The neighbour distance model was designed to evaluate the likelihood of a spatial distribution among a given group of cells. We used the level of pairwise cell overlap as the basis of the evaluation. Specifically, the level of overlap C between a pair of cells was calculated as the ratio between the distance between the two cells and the sum of their radii. The radii of cells were estimated based on the embryo volume, the total number of cells in the embryo and each cell's lineage identity, as previously described[33].

Ground truth of cell spatial distribution. The ground truth of the cell overlapping levels was collected from 50 wild-type embryos. We computed a probability density function of overlapping levels (Supplementary Fig. 5) and found that all the overlap values were between $\alpha = 0.3$ and $\beta = 0.8$.

Spatial distribution evaluation. At a given simulation step we evaluated the spatial distribution among the migrating cell and its neighbours by evaluating the level of overlap between the migrating cell and each of its neighbours. A pairwise overlap level of less than α was considered 'completely acceptable' and given a reward of 0. A pairwise overlap level greater than β was considered 'completely unacceptable' and given a reward of $-\infty$. For an overlap level between α and β, the reward r_n^i was set as a negative value of the cumulative distribution function of the overlapping value between cell i and the migrating cell, $F_C(c) = P(C \leq c)$, from the ground truth (Supplementary Fig. 5), leading to a reward in the range $[0, -1)$:

$$r_n = \begin{cases} 0 & c < \alpha \\ -F_C(c) & \alpha \leq c < \beta \\ -\infty & c \geq \beta \end{cases},$$

where $P(C \leq c)$ is the probability that C will take a value less than or equal to a predefined overlap value c. A reward was calculated between the migrating cell c_m and each of its neighbours $\{c_i; f_n(c_m, c_i) = 1$, for all $i\}$ and all the rewards R_N were summed as the total reward returned by the neighbour distance model, $R_N = \sum_i r_n^i$. If a movement action received a score of $-\infty$, the current migration epoch was stopped.

HDRL model architecture and parameters. We developed a two-level HDRL model using h-DQN[18].

Higher-level module. Network architecture. The higher-level module contains a policy network f_p^h and a CNN f_f^h. The policy network is a fully connected layer, which takes the feature vectors from the CNN to select the current subgoal $f_p^h(v_a^h) = i \in \{0, 1, \cdots, N_s\}$, where N_s is the total number of subgoal candidates. The CNN contains two convolutional layers (Supplementary Table 1), which extract feature vectors v_a^h from the annotated images x_a^h of the current time point $f_f^h(x_a^h) = v_a^h$.

Input and output. The higher-level module takes the annotated image stacks as the input. The size of the entire embryo is around 250–300 pixels. We resized the input images to 128×128. The neural network produced a pair of cells as the potential subgoal, which was sent to the lower-level module. The subgoals were selected from a candidate pool $\mathcal{S}$ that contains the secondary neighbours of the migrating cell (neighbour of a neighbour), which were determined by the neighbour relationship model f_n. $\mathcal{S} = \{(x, y) : f_n(c_m, c_i) = 1, f_n(c_i, c_z) = 1, f_n(c_m, c_x) \times f_n(c_m, c_y) = 0,$ for all i and $z = (x \text{ or } y)\}$, where c_m is the feature vector of the migrating cell (m) and $m \neq i \neq x \neq y$. c_i represents the feature vectors of the first neighbour cells (i). $c_{z=(x \text{ or } y)}$ represents the feature vectors of the cells (z) that neighbour the first neighbour cells (i). The number of subgoals was not predefined in our study; however, in our implementation, the total number of subgoals in a scenario cannot go beyond three because of the secondary neighbour rule during training.

Rewards. The rewards for achieving a subgoal and the final destination were set to 10 and 100, respectively.

Hyperparameters. The network was trained with an Adam optimizer, a batch size of 128 and a learning rate of 0.0001. A replay buffer with a size of 128 was used and training samples were stored and randomly sampled from the buffer to train the network. Similar to a previous study[7], a target network was used to stabilize the training process and its parameters were copied from the network of the higher-level module every 20 epochs. The reward discount factor γ was set to 0.8 and the ϵ greedy factor was set to 0.95. The network was trained for 300 epochs.

Lower-level module. Network architecture. The CNN f_f^l in the lower-level module has the same architecture as in the higher-level module. The policy network f_p^l is a fully connected layer, which takes the feature vectors v_a^l from the CNN to select

the current atomic movement action $f_f^l(x_a^l) = v_a^l$, $f_p^l(v_a^l) = i \in \{0, 1, \cdots, N_a\}$, where x_a^l represents the input image stacks and N_a is the total number of atomic movement actions.

Input and output. The lower-level module takes the annotated image stacks, as described in the higher-level module, with the subgoal from the higher-level module marked with yellow (Fig. 2b) as the input, and the output of the module is the atomic movement action from one of the eight directions of action in the x–y plane with 45° between each of them. The quantal step size of the movement was fixed to the average value of the step size of the migrating cell and its neighbours during the migration process with the observational data.

Rewards. The lower-level module takes rewards from both the local and long-range feedback. The local feedback consists of two parts: (1) a reward R_N according to the distribution of the normalized distance between the migrating cell and its neighbours output by the neighbour distance model (Neighbour distance model) and (2) a reward R_M, which equals 1 if the current situation is classified as a directional movement by the motion model, and otherwise is 0. For the long-range feedback, when the subgoal is achieved (that is, the migrating cell becomes a stable neighbour of the subgoal cells for 15 time steps), a reward R_L, which equals 10 is given. The total reward R is represented as $R = R_N + R_M + R_L$.

Hyperparameters. The network was trained with an Adam optimizer, a batch size of 32 and a learning rate of 0.00002. The replay buffer contained 4,000 samples and the target network was updated every 1,000 iterations. The reward discount factor γ was set to 0.98 and the ϵ greedy factor was set to 0.9.

Transferred motion model. The TMM $f_t = \{0,1\}$ was designed to classify the movement type (rosette-based versus non-rosette-based movements) discovered in the Cpaaa migration. It contains a fully connected neural network and the CNN, transferred from the trained lower-level HDRL module f_f^l.

Dataset. The image stacks used to train the TMM, the labelling and the training/test data split strategies are the same as those in the motion model. The input of the TMM is the whole embryonic image stack (see the higher-level module's input) rather than the cropped images (see the motion model's input).

Neural network architecture. The TMM consists of two parts: (1) two convolutional layers transferred from the lower-level HDRL module with the trained parameters and (2) three fully connected layers following the convolutional layers with randomly initialized weights.

Training strategy. During the training of TMM, the weights in the convolutional layers were frozen and only the weights of the fully connected layers were allowed to update. The training loss and accuracy of the TMM converged around 40 epochs, with an Adam optimizer, a mini-batch size of 10 and a learning rate of 0.0001.

Analysis of feature maps. Ablation experiments were performed as follows.

Ablated input image. An ablated input image $x_a \in \mathbb{R}^{128\times128}$ was generated by removing the cell(s) of interest (Cpaaa or its neighbour cells) during the input image-generating process. All the ablated Cpaaa's neighbour cells were identified by the neighbour relationship model.

Summary feature and difference maps. All the feature maps $F_i \in \mathbb{R}^{8\times8}$, $i = 1, 2, \cdots, N_i$ (where N_i is the total number of feature maps) for ablation experiments were generated from the CNN in the TMM at observational time steps. At each time step, we summed all 64 individual feature/difference maps to a summary feature/difference map $F = \sum_{i=1}^{64} F_i$ (Fig. 5e).

Effective area. At a given time step of an embryo, all pixels of the summary difference maps were classified into two categories according to the intensity using K-means unsupervised clustering with $K = 2$. Pixels in the cluster with higher intensity are referred to as the effective area. The numbers of pixels in the effective area of the summary difference map at each time step in three embryos were also recorded (Supplementary Fig. 4).

Ablated cell area. The ablated cell area was defined as the area in feature maps that spatially corresponds to the ablated cell(s) at a given time step of an embryo. This was calculated from an ablated cell's location and its radii in the input image (Neighbour distance model). Specifically, four bounding box locations (x, y) of each ablated cell in the input image were first downscaled by multiplying by $(\frac{8}{128}, \frac{8}{128})$ and then rounded down to obtain the mapped pixel(s) in the feature maps. The ablated cell area was obtained by aggregating all the mapped pixels of ablated cells at a given time step of an embryo.

Microscopy, cell tracking and visualization. *C. elegans* culture, microscopy and cell lineage tracing were performed as previously described[26]. The following *C. elegans* strains were used in this study: *BV24* (*ltIs44* [*pie-1p::mCherry::PH(PLC1delta1)* + *unc-119*(+)] *zuIs178* [*his-72p* (*1 kb*)*::his-72::SRPVAT::GFP::his-72 3UTR* + *unc-119*(+)] *V*) (for Cpaaa) and DCR4318 (*olaex2540* [*Punc-33_PHD_GFP_unc54*, *Punc-122_RFP*]; *ujIs113*) (for mu_int_R and CANL). Videos of the movement of Cpaaa, mu_int_R and CANL were generated using the WormGUIDES software[34] from cell tracking of a real embryo, which facilitates the visualization of spatiotemporal dynamics of selected cells in *C. elegans* embryos using 3D rendering.

Characterization of Cpaaa migration paths. The Cpaaa migration paths were represented as a time plot of the distance between Cpaaa and the target cell (ABarpaapp). The migration paths among a given group of embryos were represented as a time plot of the average migration path with one standard deviation. For paths in the observational data, a temporal alignment was applied to minimize the temporal variation of cell birth and migration events among embryos. Specifically, the middle point between the maximal and minimal distance in each migration path was used to identify a reference time point. All migration paths were aligned by the reference time point before calculating the average path and standard deviation.

Software and dataset availability. The software was constructed based on a method described in ref. [35] and implemented with Python 3 (version 3.6.2), including several packages: PyTorch[36] (version 0.2.0_3 (old) and 0.4.1), Mesa[37] (version 0.8.1), PIL[38] (version 4.2.1), scikit-learn[39] (version 0.19.1) and NumPy[40] (version 1.15.0). The computational platform used in our study was an Nvidia DGX workstation with two AMD EPYC 32-core central processing units and four Tesla V100 16-GB graphics processing units.

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

Data availability

The data that support the findings of this study have been deposited at https://drive.google.com/drive/folders/1K5DeN2oTw_KXWgtDxaRMlTrc5MS46avY?usp=sharing. A 50 wild-type *C. elegans* dataset, embryonic data for Cpaaa training and the TMM evaluation, as well the data for mu_int_R case, are included, named WT50_release, Cpaaa_release, cpaaa_1(2,3) and mu_int_R_CANL_1(2), respectively.

Code availability

Source code with data information and several pre-trained models are available at https://github.com/daliwang/hdrl4cellmigration (https://doi.org/10.5281/zenodo.543098).

Received: 8 June 2021; Accepted: 2 December 2021;
Published online: 10 January 2022

References

1. Belthangady, C. & Royer, L. A. Applications, promises and pitfalls of deep learning for fluorescence image reconstruction. *Nat. Methods* **16**, 1215–1225 (2019).
2. Moen, E. et al. Deep learning for cellular image analysis. *Nat. Methods* **16**, 1233–1246 (2019).
3. Barnes, K. M. et al. Cadherin preserves cohesion across involuting tissues during *C. elegans* neurulation.*eLife* **9**, e58626 (2020).
4. Buggenthin, F. et al. Prospective identification of hematopoietic lineage choice by deep learning. *Nat. Methods* **14**, 403–406 (2017).
5. Keller, P. J. Imaging morphogenesis: technological advances and biological insights. *Science* **340**, 1234168 (2013).
6. Ladoux, B. & Mège, R.-M. Mechanobiology of collective cell behaviours. *Nat. Rev. Mol. Cell Biol.* **18**, 743–757 (2017).
7. Mnih, V. et al. Human-level control through deep reinforcement learning. *Nature* **518**, 529–533 (2015).
8. Lillicrap, T. P. et al. Continuous control with deep reinforcement learning. In *Proc. 4th International Conference on Learning Representations* (eds Bengio, Y. & LeCun, Y.) 1–10 (ICLR, 2016).
9. Silver, D. et al. Mastering the game of Go with deep neural networks and tree search. *Nature* **529**, 484–489 (2016).
10. Silver, D. et al. A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play. *Science* **362**, 1140–1144 (2018).
11. Gu, S., Holly, E., Lillicrap, T. & Levine, S. Deep reinforcement learning for robotic manipulation with asynchronous off-policy updates. In *Proc. IEEE International Conference on Robotics and Automation* (eds Chen, I. M. & Ang, M.), 29-3 (ICRA, 2017).
12. Nguyen, H. & La, H. Review of deep reinforcement learning for robot manipulation. In *Proc. 3rd IEEE International Conference on Robotic Computing* (eds Brugali, D., Sheu, P. C.-Y., Siciliano, B. & Tsai, J. J. P.) 590–595 (IEEE, 2019).
13. Kalashnikov, D. et al. Scalable deep Reinforcement learning for vision-based robotic manipulation. In *Proc. 2nd Annual Conference on Robot Learning* Vol. 87 (eds Billard, A. & Siegwart, R.) 651–673 (2018).

14. Arulkumaran, K., Deisenroth, M. P., Brundage, M. & Bharath, A. A. Deep reinforcement learning: a brief survey. *IEEE Signal Process. Mag.* **34**, 26–38 (2017).
15. Neftci, E. O. & Averbeck, B. B. Reinforcement learning in artificial and biological systems. *Nat. Mach. Intell* **1**, 133–143 (2019).
16. Sutton, R. S., Precup, D. & Singh, S. Between MDPs and semi-MDPs: a framework for temporal abstraction in reinforcement learning. *Artif. Intell.* **112**, 181–211 (1999).
17. Vezhnevets, A. S. et al. FeUdal networks for hierarchical reinforcement learning. In *Proc. 34th International Conference on Machine Learning*, *ICML 2017* Vol. 70 (eds Precup, D. and Teh, Y.) 3540–3549 (ACM, 2017).
18. Kulkarni, T. D., Narasimhan, K. R., Saeedi, A. & Tenenbaum, J. B. Hierarchical deep reinforcement learning: integrating temporal abstraction and intrinsic motivation. In *Proc. 30th International Conference on Neural Information Processing Systems* (eds Lee, D. & Sugiyama, M.) 3682–3690 (ACM, 2016).
19. Tessler, C., Givony, S., Zahavy, T., Mankowitz, D. J. & Mannor, S. A deep hierarchical approach to lifelong learning in minecraft. In *Proc. 31st AAAI Conference on Artificial Intelligence*, *AAAI 2017* (ed. Zilberstein, S.) 1553–1561 (ACM, 2017).
20. Sulston, J. E., Schierenberg, E., White, J. G. & Thomson, J. N. The embryonic cell lineage of the nematode *Caenorhabditis elegans*. *Dev. Biol.* **100**, 64–119 (1983).
21. Bao, Z. et al. Automated cell lineage tracing in *Caenorhabditis elegans*. *Proc. Natl Acad. Sci. USA* **103**, 2707–2712 (2006).
22. Santella, A., Du, Z., Nowotschin, S., Hadjantonakis, A. K. & Bao, Z. A hybrid blob-slice model for accurate and efficient detection of fluorescence labeled nuclei in 3D. *BMC Bioinformatics* **11**, 580 (2010).
23. Santella, A., Du, Z. & Bao, Z. A semi-local neighborhood-based framework for probabilistic cell lineage tracing. *BMC Bioinformatics* **15**, 217 (2014).
24. Katzman, B., Tang, D., Santella, A. & Bao, Z. AceTree: a major update and case study in the long term maintenance of open-source scientific software. *BMC Bioinformatics* **19**, 121 (2018).
25. Wang, Z. et al. Deep reinforcement learning of cell movement in the early stage of C. *elegans* embryogenesis. *Bioinformatics* **34**, 3169–3177 (2018).
26. Shah, P. K. et al. PCP and SAX-3/Robo pathways cooperate to regulate convergent extension-based nerve cord assembly in *C. elegans*. *Dev. Cell* **41**, 195–203.e3 (2017).
27. Moore, J. L., Du, Z. & Bao, Z. Systematic quantification of developmental phenotypes at single-cell resolution during embryogenesis. *Development* **140**, 3266–3274 (2013).
28. Paré, A. C. et al. A positional Toll receptor code directs convergent extension in *Drosophila*. *Nature* **515**, 523–527 (2014).
29. Du, Z. et al. The regulatory landscape of lineage differentiation in a metazoan embryo. *Dev. Cell* **34**, 592–607 (2015).
30. Hunter, C. P. & Kenyon, C. Spatial and temporal controls target *pal-1* blastomere-specification activity to a single blastomere lineage in *C. elegans* embryos. *Cell* **87**, 217–226 (1996).
31. Wu, Y. et al. Inverted selective plane illumination microscopy (iSPIM) enables coupled cell identity lineaging and neurodevelopmental imaging in *Caenorhabditis elegans*. *Proc. Natl Acad. Sci. USA* **108**, 17708–17713 (2011).
32. Banino, A. et al. Vector-based navigation using grid-like representations in artificial agents. *Nature* **557**, 429–433 (2018).
33. Wang, Z., Li, H., Wang, D. & Bao, Z. Cell neighbor determination in the metazoan embryo system. In *Proc. 8th ACM International Conference on Bioinformatics*, *Computational Biology and Health Informatics* (eds Haspel, N. and Cowen, L.) 305–312 (ACM, 2017).
34. Santella, A. et al. WormGUIDES: an interactive single cell developmental atlas and tool for collaborative multidimensional data exploration. *BMC Bioinformatics* **16**, 189 (2015).
35. Wang, Z. et al. An observation-driven agent-based modeling and analysis framework for *C. elegans* embryogenesis. *PLoS ONE* **11**, e0166551 (2016).
36. Paszke, A. et al. in *Proc. NeurIPS* Vol. 32 (eds Wallach, H. et al.) 8024–8035 (NIPS, 2019).
37. Kazil, J., Masad, D. & Crooks, A. *Utilizing Python for Agent-based Modeling: the Mesa Framework* Vol. 12268 (eds Thomson, R. et al.) 308–317 (Lecture Notes in Computer Science, Springer, 2020).
38. Umesh, P. Image processing in Python. *CSI Commun.* **23**, http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.362.4331&rep=rep1&type=pdf#page=25 (2012).
39. Pedregosa, F. et al. Scikit-learn: machine learning in Python. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011).
40. Harris, C. R. et al. Array programming with NumPy. *Nature* **585**, 357–362 (2020).

Acknowledgements

We thank A. Santella for discussions and technical help and H. Shroff and Q. Morris for critiquing the manuscript. This study was partly supported by an NIH grant (R01GM097576) to Z.B. and D.W. Research in Z.B.'s laboratory is also supported by an NIH centre grant to MSKCC (P30CA008748). This research used resources of the Compute and Data Environment for Science (CADES) at the Oak Ridge National Laboratory, which is supported by the Office of Science of the US Department of Energy under contract no. DE-AC05-00OR22725.

Author contributions

Z.W., Y.X., D.W. and Z.B. designed the experiments. Z.W., J.Y. and Y.X. performed the experiments and analysed the data. Z.W., Y.X., D.W., J.Y. and Z.B. wrote the manuscript. D.W. and Z.B. supervised the project.

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/s42256-021-00431-x.

Correspondence and requests for materials should be addressed to Dali Wang or Zhirong Bao.

Peer review information *Nature Machine Intelligence* thanks Nico Scherf and the other, anonymous, reviewer(s) for their contribution to the peer review of this work.

Reprints and permissions information is available at www.nature.com/reprints.

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

This is a U.S. government work and not under copyright protection in the U.S.; foreign copyright protection may apply 2022

nature portfolio

Corresponding author(s): DBPR

Last updated by author(s): Nov 1, 2021

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	The following software is used: AceTree.
Data analysis	The following software is used: Pytorch 0.4.1, Mesa 0.8.1, PIL 4.2.1, scikit-learn 0.19.1, Numpy 1.15.0

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

The data that support the findings of this study have been deposited in https://drive.google.com/drive/folders/1K5DeN2oTw_KXWgtDxaRMlTrc5MS46avY?usp=sharing. 50 wild type C. elegans dataset, embryonic data for Cpaaa training and the TMM evaluation, as well the data for mu_int_R case are included, named WT50_release, Cpaaa_release, cpaaa_1(2,3), and mu_int_R_CANL_1(2), respectively.

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.*

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
☒	☐ Human research participants
☒	☐ Clinical data
☒	☐ Dual use research of concern

Methods

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