DeXtrusion: automatic recognition of epithelial cell extrusion through machine learning in vivo

Alexis Villars, Gaëlle Letort, Léo Valon, Romain Levayer

DeXtrusion: Automatic recognition of epithelial cell extrusion through machine learning *in vivo*

Alexis Villars[1,2*], Gaëlle Letort[1*], Léo Valon[1] and Romain Levayer[1§]

1. Department of Developmental and Stem Cell Biology, Institut Pasteur, Université de Paris Cité, CNRS UMR 3738, 25 rue du Dr. Roux, 75015 Paris, France

2. Current address: Department of Biosystems Science and Engineering, ETH Zürich, Mattenstrasse 26, 4058 Basel, Switzerland

* These authors contributed equally to this work

§ Lead contact, correspondance to: romain.levayer@pasteur.fr

Keywords: Deep-learning, convolutional neural network, epithelia, live imaging, extrusion, cell division, SOPs, *Drosophila*

Summary statement

We describe a new and fully available method based on machine-learning to detect automatically cellular events (including cell extrusion, cell division and cell differentiation) from fluorescent movies of epithelia *in vivo*.

Abstract

Accurately counting and localising cellular events from movies is an important bottleneck of high content tissue/embryo live imaging. Here, we propose a new methodology based on deep learning allowing automatic detection of cellular events and their precise x-y-t localisation on live fluorescent imaging movies without segmentation. We focused on the detection of cell extrusion, the expulsion of dying cells from the epithelial layer, and devised DeXtrusion: a pipeline based on recurrent neural networks for automatic detection of cell extrusion/cell death events in large movies of epithelia marked with cell contour. The pipeline, initially trained on movies of the *Drosophila* pupal notum marked with fluorescent E-cadherin, is easily trainable, provides fast and accurate extrusion predictions in a large range of imaging conditions, and can also detect other cellular events such as cell division or cell differentiation. It also performs well on other epithelial tissues with reasonable retraining. Our methodology could easily be applied for other cellular events detected in live fluorescent microscopy and help to democratise the use of deep learning for automatic event detections in developing tissues.

Introduction

Epithelial tissues can be dramatically remodelled during embryogenesis or in adult organs undergoing fast turnover. This is often associated with high rates of cell elimination, which requires the fine control of the absolute number of dying cells, as well as their distribution in time and space. Cell extrusion is a sequence of remodelling steps leading to apical constriction and cell elimination from the epithelial layer without impairing the sealing properties of the tissue (Rosenblatt et al., 2001). This process is highly coordinated between the extruding cell and its neighbours: as the cell extrudes its neighbours are brought close to each other to maintain epithelial sealing and stability (Villars and Levayer, 2022).

The tight spatiotemporal control of epithelial cell apoptosis plays an essential role during tissue morphogenesis (Ambrosini et al., 2017). For instance, a spatial bias in the distribution of cell death can locally modulate growth and final tissue shape (Matamoro-Vidal et al., 2022), apoptosis and cell extrusions can generate local traction forces to fuse tissues (Toyama et al., 2008), promote locally tissue bending (Monier et al., 2015; Roellig et al., 2022) or can be permissive for global tissue remodelling through the modulation of tissue viscosity (Ranft et al., 2010; Suzanne et al., 2010). Moreover, the tight regulation of the precise spatiotemporal distribution of cell extrusion/cell death is also essential to maintain the cohesion of the tissue, especially in conditions with high rates of cell elimination (Valon et al., 2021). These examples all rely on the precise regulation of the number and spatiotemporal localisation of dying cells. Yet, despite the fast progress in our understanding of the molecular regulators of programmed cell death and extrusion, we still fail to predict when, where and how many cells will die in a tissue. This most likely relies on the multiple feedback that can modulate death rate/cell extrusion at various spatial and temporal scales (Villars and Levayer, 2022), which includes the activation of the pro-survival signal ERK in the neighbouring cells (Bock et al., 2021; Gagliardi et al., 2021; Valon et al., 2021), long-range coordination for cell extrusion (Aikin et al., 2020; Takeuchi et al., 2020) as well as positive feedbacks on apoptosis (Perez-Garijo et al., 2013). Thus, obtaining a comprehensive understanding of epithelial cell death regulation entails the dissection of multi-layered regulations integrating feedback on several spatial and temporal scales as well as several steps of decisions. Such a challenging question requires a highly quantitative dataset on the total number of cell death as well as their precise spatiotemporal distribution and high throughput methods to compare these values in various perturbed backgrounds.

The recent advances in long-term live imaging provide a wealth of data regarding tissue dynamics, especially for epithelia in 2D. However, retrieving cellular events quantitatively in a high throughput manner remains extremely challenging. For instance, there are more than one thousand extrusions, distributed all over the epithelium in the *Drosophila* pupal notum in less than 12 hours (Guirao et al., 2015; Valon et al., 2021). So far, quantitative analyses of cell death/extrusion were performed using laborious manual detection of these events (Moreno et al., 2019; Valon et al., 2021; Villars et al., 2022). This highly time-consuming task remains one

of the main bottlenecks for comparing a high number of conditions with precise quantitative readouts. Alternatively, automatic epithelial cell death detection was performed through systematic segmentation and tracking of all the cells (Etournay et al., 2016; Guirao et al., 2015). However, this method typically entails extensive manual corrections even when using machine learning-enhanced segmentation (Aigouy et al., 2020), and still represents quite an important load of work for large fields of view and long timescales, hindering large-scale analysis on many tissues.

Altogether, these challenges and needs call for an automatic tool that would allow accurate spatiotemporal detection of cellular events without relying on systematic segmentation and tracking of cells. The recent progress of computer vision opened the possibility to automatise the detection of objects and patterns in biological images and image series (Hallou et al., 2021). In particular, deep learning approaches were used successfully to recognise cellular events such as cell division and cell death in yeast (Aspert et al., 2022) or in mammalian cell culture (Kabir et al., 2022; La Greca et al., 2021; Mahecic et al., 2022; Phan et al., 2019; Shkolyar et al., 2015). However, this was mostly applied to transmission light microscopy and these pipelines are not applicable to large samples and embryos, where imaging mostly relies on fluorescent and confocal microscopy. As such, there is currently to our knowledge no implemented solution for the automatic detection of cell death and cell extrusion events from epithelia *in vivo*.

To answer these challenges, we devised here a supervised machine-learning pipeline called DeXtrusion. The detection of extrusion is performed by screening the entire movie with sliding windows and detecting cellular events on each window. The core of our pipeline is based on a recurrent neural network which classifies each image sequence (called DeXNet). These local classifications are then post-processed together at the movie level to convert them to probability maps of the presence of a cellular event and eventually precise single-point event detection. We devised and applied this method on the *Drosophila* pupal notum, a single-layer epithelium, using cell-contour-labelled epithelia with tagged E-cadherin. The method is flexible enough to provide accurate and precise predictions with movies of different temporal resolutions, pixel sizes, imaging set-ups as well as different E-cadherin labelling, without any need for segmentation. Moreover, DeXtrusion generalises well out-of-the-box to other epithelia (e.g.: pupal abdomen, pupal wing), which can be further

enhanced by retraining on small data sets. The same methodology can also detect other cellular events such as cell division and cell differentiation. By resolving the bottleneck of extrusion/cell death detection, DeXtrusion will open the way for a more systematic and quantitative characterisation of cell death distribution in large datasets which will be essential to understand its multi-layered regulation. The same methodology could easily be applied to any cellular event detected in fluorescent live imaging. DeXtrusion is distributed as a Python module, available open source on GitLab (https://gitlab.pasteur.fr/gletort/dextrusion) along with our trained neural networks and scripts (Jupyter notebooks and Fiji macros) to facilitate its usage, our annotated datasets used for training of the pipeline are available on Zenodo (Villars et al., 2023) (https://doi.org/10.5281/zenodo.7586394).

Results

DeXNet: a neural network to recognise extrusion in cropped image sequences.

We first aimed at detecting cell extrusion in the *Drosophila* pupal notum: a single-layer epithelium on the back of the developing *Drosophila* pupal thorax with a high rate of cell extrusion/apoptosis that follows stereotypical patterns (**Fig.1A**) (Guirao et al., 2015; Levayer et al., 2016; Marinari et al., 2012; Valon et al., 2021; Villars et al., 2022). We used a dataset generated in the laboratory covering a large number of extrusion events (6700 in the training set and 2320 in the test set see **Supplementary tables 1 and 2**). This dataset was made of large-scale movies of the pupal notum obtained from two different imaging set-ups (see **Material and Methods**), with different frame rates, signal-to-noise ratios, and different fluorescent proteins coupled to the adherens junction protein E-cadherin. It contains both Wild Type (WT), mutants and drug-perturbed conditions to obtain a model robust in a large range of conditions (see **Supplementary tables 1 and 2**). Cell extrusions were manually annotated by clicking on the point of termination of apical constriction while control positions were drawn randomly, excluding locations containing an extrusion (see **Material and Methods** for details regarding events detection). To handle data with different frame rates and spatial resolutions, we fixed a reference spatiotemporal scale (0.275 microns/pixel and 5 min/frame) on which the neural network was trained. Every processed movie is rescaled to this referential before being processed.

To detect with spatiotemporal accuracy all cell extrusions, we generated a pipeline called DeXtrusion, which screens through the entire movie using overlapping sliding windows and assigns a probability of extrusion for each cropped sequence. DeXtrusion refers to the full pipeline detecting cellular events from the full movies (see below and **Figure 2**), while DeXNet refers to the neural network used to classify each cropped image sequence. Each cropped image sequence (a series of 10 images, one every 5 minutes, of 45x45 pixels in the reference scale, representing 2-3 cells per frame) was processed through a neural network called DeXNet to estimate the presence or absence of an extrusion event (**Fig.1B**). For this, we built and trained a neural network to classify each image sequence (**Fig.1C-C'**). We first chose an architecture based on Convolutional Neural Network (CNN), efficient for image-based classification (LeCun et al., 2015). Since cell extrusion is mostly defined by the dynamics of cell shape changes (progressive apical constriction), we decided to include temporal information by using a recurrent neural network architecture. We chose a Gated Recurrent Unit (GRU) architecture which computes and propagates temporal information while preserving a parsimonious network architecture (Cho et al., 2014). Each image of the temporal sequence is first encoded using the CNN (**Fig.1C'**) and reduced in a vector of discriminant features (64), which is then combined with the other temporal image feature vectors into the GRU (**Fig.1C**). The output of the time distributed GRU layer then goes through a sequence of densely connected layers that perform the final classification (**Fig.1C purple**). Eventually, the network provides a probability of containing an extrusion for each cropped image sequence. This probability was finally thresholded to a binary output (extrusion or control event). We trained DeXNet on image sequences extracted from our dataset, split in a training and test set (25 versus 7 movies, see **Material and Methods**). We took cropped image sequences in the reference scale centred spatially and temporally around the manually detected extrusions (time 0 being defined by the termination of apical constriction, **Fig. 1B**), while control sequences were made of similar cropped image sequences that do not contain any event (see **Material and Methods**). To avoid imbalanced detection in our training dataset (due to the over-representation of "no event" sequences relative to extrusion), we have equalised the number of extrusion and control sequences used for the training (see **Material and Methods**). The size of the temporal window, the x-y size of the cropped input images, as well as other hyperparameters of the model (e.g: number of epochs -the number of times the dataset is processed by the network for

training-, of features, data augmentation), were tuned and optimised by testing several sets of values (**Fig. S1, S2 and S3**). The parameters were compared after processing through the full DeXtrusion pipeline (see below and **Figure 2**) by comparing the precision (number of true positives out of all detected events), the recall (proportion of manually annotated cellular events detected by the pipeline), the computation time, and the F1 score (a measurement of quality and exhaustivity of detections, see **Material and Methods**). For instance, the temporal size of the sequences (10 frames, 50 minutes) was chosen so that it contains enough information to recognise extrusion (which lasts on average 20 to 30 minutes (Villars et al., 2022)) while reducing the chance to capture two events and reducing calculation time (see **Material and Methods**, **Fig. S1A-D**). Importantly, we observed a drastic reduction of precision and recall when restricting the time windows to two frames, illustrating the key role of temporal information for accurate detection (**Fig. S1 A-D**). Taken together, the selected hyperparameters allowed the model to converge properly under 50 epochs and took around 55 min of training with 1 GPU (**Fig. 1D**). Compared to our manual annotation, the classification of the neural network on cropped image sequences extracted from our test dataset resulted in an accuracy of 97% (proportion of correct predictions out of all predictions, see **Material and Methods**), with 2% False Positive (detected extrusions which are not real extrusions) and 1% False Negative (real extrusions that are not detected) (**Fig.1E-F**). Of note, these 2% of false positives could constitute a challenge for detection in the full movies since the proportion of “no event” sequences is much higher than the proportion of real extrusions. This may generate a significant proportion of false positives (thus reducing our “precision”). We will discuss the optimisation of the precision in later sections.

DeXtrusion: from cropped images classification to whole-movie extrusion detection

So far, our trained neural network DeXNet provides a fast and efficient classification of each image sequence. To detect extrusion events across the tissue and at all temporal time points, the complete movie was split into 45x45 pixels cropped images with 50% overlap. The time axis was divided into sequences of 10 frames every two time-points after rescaling the time to the reference scale (**Fig. 2A**). These parameters are a good compromise between the spatiotemporal precision of the results and the speed of computation and can be tuned by users. For each sliding window, DeXNet provides a

single probability of extrusion. This value is then allocated in the original movie in a 22x22 pixels and 5-time points regions around the centre of the cropped sequence. The final local probability output is then obtained by averaging for each pixel the probability of the different overlapping windows (in time) generating an extrusion probability map on the whole movie with a spatial resolution of 22 pixels in the reference scale (**Fig. 2A,B, Movie 1**). This output, once rescaled back to the original movie spatio-temporal scale, can be directly used to screen visually the detection of extrusion events and assess putative tissue patterns (**Fig. 2B**).

To count the number and precisely localise cell extrusion events, this probability map must be converted to single point event detection. Single events in our probability map are associated with high probability within several consecutive time windows and on a given x-y surface, thus generating a volume (x-y-t) of pixels with high probability (**Fig. 2A,C, Movie 1**). We first thresholded the probability map to obtain a 3D mask of positive detections. To filter out false positives which are usually detected on smaller areas and fewer time windows (**Fig. 2C,C'**), we also thresholded the size of the positive volume to fit the minimal volume associated with extrusion events (which last between 20 and 30 minutes and should cover at least one cell diameter (Villars et al., 2022)). The probability threshold and volume threshold were optimised to maximise the precision and recall (**Fig. S2, Material and Methods**). We also used a watershed separation to separate close events. Eventually, the exact location of the extrusion is defined by the centroid of the high-probability volume. Detected extrusions are then exported as a list of point ROIs (Region Of Interest) compatible with Fiji ROI manager (Schindelin et al., 2012) (**Fig. 2A**).

To assess the performance of DeXtrusion, we compared the resulting ROIs with the manually annotated ROIs of our test dataset. We first measured systematically the x-y Euclidean distance and time distance between manually annotated and automatically detected extrusions (**Fig. 2D,D'**). This showed that 70.6% of the detections are below a 15-pixels distance (4,12 μm ~ 1 cell radius) and +/- 4 time-frames shift (+/- 20 minutes). We then classified detections as correct for spatial distance below 15 pixels and below 4 time-frames distance between the DeXtrusion and manually located ROIs. Doing so, we obtained a recall of 0.87 (proportion of manually annotated cellular events detected by the pipeline) and a precision of 0.46 (proportion of detected events that are indeed extrusion, averaged on 5 independent trained networks). We will discuss

the optimisation of the precision in the next section. Finally, we measured systematically the total number of detected extrusions compared to manually annotated events at every developmental time to check whether the accuracy of our detection is sensitive to the developmental stage. We observed a fairly constant error (~3 errors every 10 frames/50 minutes) suggesting that our methodology is robust at all developmental stages imaged in the notum (**Fig. 2E-F**). Thus, our methodology can retrieve the vast majority of the extrusion events at any stage of development.

Optimisation of the model for the detection of extrusion

So far, we have designed and optimised DeXtrusion to detect exhaustively all the extrusion events, which led to the high recall of 0.87 (87% of events detected). However, we have not yet fully optimised the methodology to filter out false positive detections (to optimise the so-called precision). Indeed, many cellular events which share some phenotypic similarities with extrusion are very frequently miss labelled as extrusion (**Fig. 3 A-D**, **Movie 1**). This includes, for instance, the formation of Sensory Organ Precursors (SOPs) which through asymmetric cell division form cells with very small apical areas (Gho et al., 1999) which are often detected as extrusions (**Fig. 3B**). Similarly, the shrinkage of cell apical area concomitant with cytokinesis and furrow formation are frequently misclassified as extrusions (**Fig. 3C**). Thus, to enhance the precision of DeXtrusion (the proportion of correct extrusion detection), we decided to add these other cellular events in DeXNet to discriminate them from extrusion. We trained the network to detect cell extrusions, cell divisions and SOPs or the absence of events using manually annotated events in our dataset (6700 extrusions, 3021 divisions and 3054 SOPs, **Fig. 3F**). DeXNet was able to categorise the cropped image sequences into these four classes with high accuracy on our training set (0.986, 0.989 and 0.976 for respectively extrusions, SOPs and divisions, **Fig. 3G, Movie 1**). By including this new DeXNet in our pipeline, we significantly increased the precision of DeXtrusion (from 0.26 to 0.41, single run with the selected networks having the highest recall, **Fig. 3E,E''**), while this did not impact significantly the recall (**Fig. 3E'**). This is also reflected by the F1 score (**Fig. 3E**), a measurement of quality and exhaustivity of detections (see **Material and Methods,** from 0.40 to 0.54). To further enhance our precision, we then manually screened all remaining false positive detections and included image sequences representative of the patterns of these false positives in the training inputs as control sequences and retrained the DeXNet network (using the

network including the four classes of events). This reinforcement increased the model's precision to 0.52 for similar recall (0.92 compared to 0.90). Finally, we exploit the inherent stochasticity of the training process (which always leads to different network weights and biases) to confront the predictions from two independently trained networks (Segebarth et al., 2020). Averaging these two independent classifications had the most significant effect and increased the precision to 0.72 (recall of 0.86 and f1-score of 0.78). This barely affected the recall (**Fig.3E-E''**, 2nets), but it increased the calculation time by a factor of two. Of note, the accuracy was much lower for the test movie of the colcemid injected pupae (**Fig. 3E-E''**, deep blue, movie ID18), whose phenotype was poorly represented in the training dataset and corresponded to an extreme condition where the mode of extrusion and the architecture of the tissue are quite different (Villars et al., 2022). After all these optimisation steps, the final model (2nets) can still detect all cellular events on a movie covering the full pupal notum (1200 by 1200 pixels 200 time points) in less than 40 min on a regular PC with 1 GPU. Taken together, these optimisations steps led to a drastic increase in precision while maintaining similar recall on test movies. The last remaining false positive (~20%) can be easily filtered out manually from the final ROIs list. We estimated that the full procedure (DeXtrusion computation time+manual correction) takes between 1 or 2 hours for a movie covering the full pupal notum over 200 times (representing roughly 1000 extrusions). This corresponds to a drastic time saving compared to the exhaustive manual detection of extrusions (~10 hours for a trained user (Valon et al., 2021) versus approximately one hour of manual correction with DeXtrusion).

Generalisation of DeXtrusion to mutant contexts and other epithelia

The performance of a neural network is usually highly limited to the training data range and cannot easily generalise to other conditions/tissues not represented in the training dataset (Mockl et al., 2020). To test the capacity of generalisation of DeXtrusion, we challenged our pipeline by testing its performance on different data from relatively similar biological conditions to very different biological contexts. We first trained two "full" networks, using all our annotated data (training+test dataset). To adapt DeXtrusion to other tissues (**Fig. 4A**) where the cell size and duration of cellular events are not of the same scale, the reference scale at which movies are resized is calculated so that cell diameter is around 25 pixels and extrusion taking place over 4-5 frames. We first tested DeXtrusion on *Drosophila* pupal notum depleted for EGFR (UAS-

EGFR-dsRNA driven by pnr-Gal4), a condition that modifies tissue shape and the spatiotemporal distribution of extrusion while not affecting so much the extrusion process per se (Moreno et al., 2019; Valon et al., 2021). We obtained an overall good detection level in this context (**Fig. 4A,B,B'**,**E Movie 2**). Since annotated data do not exist for these movies, we randomly sampled the prediction results to manually check the events (see **Material and Methods**). Using this method, we estimated that 87% of extrusion detections were correct (precision=0.87, **Fig. 4E**). We further challenged DeXtrusion using the *Drosophila* pupal wing, an epithelial with similar cell shape and size to the pupal notum (Aigouy et al., 2010; Etournay et al., 2015; Farhadifar et al., 2007) (**Fig. 4A,C,C'**, **Movie 3**). Using a previously published movie of E-cad::GFP pupal wing (Etournay et al., 2015), we obtained a precision of 0.786 (excluding ROIs outside the wing, see **Material and Methods**, **Fig. 4E**) and an estimated recall of 0.85 (estimated on 59 extrusions manually annotated). These scores could be significantly improved by retraining DeXNets on a small proportion of manually annotated extrusions and divisions from the pupal wing (75 events), reaching a precision of 0.91 and an estimated recall of 0.83, **Fig. 4E**, **Movie 3** and **Material and Methods**).

We further challenged DeXtrusion by testing its performance on a squamous epithelial, the larval epithelial cells of the pupal abdomen, where cells have very different shapes from the pupal notum and where extrusions occur through slightly different mechanisms (Hoshika et al., 2020; Michel and Dahmann, 2020; Teng et al., 2017; Villars et al., 2022) (**Fig. 4 A,D,D'**). This test was conducted on 4 movies from (Davis et al., 2022; Tapon and Salbreux, 2022) focusing exclusively on the larval epidermal cells. After proper rescaling, we obtained a precision of 0.737 but obtained a low recall of 0.38 (using available segmentation and annotation (Davis et al., 2022; Tapon and Salbreux, 2022), despite adjusting the threshold distances (spatial and temporal) to the scale of the movie (see **Material and Methods**). This may reflect the difference in cell morphology (long and curved junctions) and dynamics of extrusion (progressive loss of E-cad and rounding, (Teng et al., 2017) , **Fig.4 D'**). To adapt DeXtrusion to these cells, we retrained our DeXNet models using the annotated extrusions of one of the four movies. The precision increased to 0.857 and the recall nearly doubled to 0.69 (**Fig. 4E**, **Movie 4**). Increasing the retraining data to 2 movies continued to improve the performance but less drastically (precision of 0.86 and a recall of 0.74).

Altogether, this demonstrates that DeXtrusion can robustly detect extrusion events on various tissues and conditions. For situations where cell shape and the profile of extrusions are very different, few additional trainings are sufficient to reach back good precision and recall, illustrating the adaptability of DeXtrusion.

Discussion

DeXtrusion allows fast and precise detection of cell extrusions in a wide range of imaging conditions without any need for segmented dataset and can also detect accurately other cellular events (SOPs, cell divisions). While our precision (~0.8) still requires a manual correction phase to filter out false positives, this leads to a considerable gain of time compared to the manual annotation of large movies (from 10 hours to 1 hour). This opens new opportunities for systematic quantification of the spatiotemporal distribution of cell death in a large number of movies. DeXtrusion offers the possibility to screen for drugs/mutations affecting the spatiotemporal distribution of cell death and could lead to the identification of new biological factors modulating cell apoptosis and new spatiotemporal feedbacks. While we extended DeXtrusion to detect cell divisions and SOPs with the aim of improving our precision, these additional features can also be handy to analyse the spatiotemporal interplay between these three cellular events, for instance regarding the coupling between cell death and cell division and compensatory proliferation (Fan and Bergmann, 2008; Kawaue et al., 2021; Mesa et al., 2018). Combining such large datasets with graph neural network (Yamamoto et al., 2022) and spatiotemporal point pattern analysis and modelling (Valon et al., 2021) may eventually help to rapidly detect new spatiotemporal couplings at various x-y-t distances. Note however that we did not estimate the precision/recall of our pipeline to detect these other cellular events as it was not the initial scope of our study.

We challenged DeXtrusion to detect extrusions in other tissues and/or new genetic backgrounds and demonstrated how easily it could be adjusted using, when necessary, minimal retraining datasets (by manually detecting few events and without need for segmentation). This also included epithelia with very different architectures (cuboidal in the notum, squamous in larval epidermal cells). We have not tested the performance of DeXtrusion on tissue with different labelling (e.g.: membrane, actin), yet a reasonable retraining may be enough to readjust the pipeline for a wide range of

tagged proteins. Alternatively, integrating purely geometrical features in the training dataset (through segmentation of a large number of extrusions/divisions/SOPs and integrating skeletonised sequences in the training) may help to build a more generalist model less sensitive to the tagged protein and the variation of fluorescence intensity. While collecting enough segmented data may be time-consuming, it could be a very promising alternative method to build a very generalist model. Introducing manually defined features into the neural network could also be an interesting approach to increase the network performance (Huang et al., 2022) but would require more computation to extract the hand-crafted features. While we have limited our detections to extrusions, divisions and SOPs, the same pipeline could easily be trained to recognise new cellular events. Of note, the detection of cell extrusion is particularly challenging since the deformations associated with cell extrusion are not so stereotypical: some cells constrict isotropically and form clear rosettes, others constrict while elongating and losing progressively cell-cell junctions (Levayer et al., 2016; Marinari et al., 2012; Villars et al., 2022), as opposed for instance to cell divisions which are always preceded by cell rounding. We are therefore quite confident that our methodology could be very performant on a wide range of epithelial events. For instance, a significant proportion of false positives were related to local cell rearrangements and T1 transitions (Guirao and Bellaiche, 2017). Adding this new feature could not only increase the precision of our pipeline but also offer an interesting tool for large-scale analysis of tissue dynamics and the evolution of fluidity in time and space (Tetley et al., 2019). Very interestingly, Gallusser et al. (Gallusser et al., 2023) proposed recently a self-supervised pipeline to capture automatically the presence of an event based on their asymmetric signature in time. Filtering the movie to keep only potential events with their framework before to perform the event classification through DeXNet could be an additional boost to our pipeline.

The workflow that we developed here for extrusion detections can be applied to any other biological event detection, provided that enough training data are available, without the need for segmentation. Moreover, the high rate of false positives of our initial pipeline revealed the difficulty of detecting one particular type of un-frequent event in a movie containing potentially other events. The strategies we applied to improve the precision (classifying other cellular events as well, reinforcement of the training by including other typical false positives, e.g. transient cell deformations, and

averaging the prediction of several networks) led to a drastic increase in the prediction precision (~3-folds increase). To our knowledge, these methods were not so commonly used before in deep learning context and could in principle be applied to any other machine learning pipeline. By distributing DeXtrusion open-source, we offer an optimised pipeline that can be easily tuned to detect biological events in temporal series without segmentation and tracking.

Material and Methods

Generation of the training dataset

To create an annotated dataset, we used movies that had been previously manually annotated for extrusions in the laboratory for other studies (Moreno et al., 2019; Valon et al., 2021; Villars et al., 2022). Pupae were dissected and imaged on a confocal spinning disc microscope (Gataca systems) with a 40X oil objective (Nikon plan fluor, N.A. 1.30) or 100X oil objective (Nikon plan fluor A N.A. 1.30) or a LSM880 equipped with a fast Airyscan using an oil 40X objective (N.A. 1.3), Z-stacks (1 µm/slice). All the movies were built using a local z-projection plugin which follows tissue curvature (Herbert et al., 2021). We pooled together 32 movies, all from the *Drosophila* pupal notum, but with different acquisition setups, genetic backgrounds and junctions staining. The full list of movies and their characteristics is given in **Supplementary tables 1 and 2**. We split the annotated data into two datasets, one for training (**Supplementary table 1**) and a smaller one for testing (**Supplementary table 2)**. For this, we selected randomly one movie of each of our different conditions for the test set, and all remaining movies of the same condition were used for training. We obtained two datasets: the training set of 25 movies, with 6692 annotated extrusion events from 8 different conditions, and a test set of 7 movies, with 2320 annotated extrusions from 7 different conditions. These data with the manual annotation are freely available on a repository (https://doi.org/10.5281/zenodo.7586394) (Villars et al., 2023). Detection of cellular events (extrusion, SOPs, cell division and control regions) were performed manually on Fiji using dot ROIs and extracting x,y,t coordinates. For extrusion, the point of termination of apical constriction was manually clicked. For cell division, the center of the dividing cell just prior to cytokinesis (beginning of furrow constriction) were clicked. For SOPs, no specific time points were picked and all the

SOPs stage were randomly selected (from the first asymmetric division events to the late SOPs) by clicking in the center of the cell with the smallest apical area.

For each network training, the training dataset was split in 75-25% between training and validation data. Note however that these two subsets were not fully independent as they are composed of windows extracted from the same movies (training movies).

Training image sequence generation

To generate training image sequences from the ROI files and not keep all the movies in the running memory, we implemented a movie generator (MovieGeneratorFromRoi.py). It randomly selects the ROI from the input file (after manual detection of events on Fiji, see above) and saved a cropped image sequence centred around that point, with a small spatial and temporal random shift (so that the event is not perfectly centred in the window to limit a bias toward the centre of the image). For control (no event) windows, positions are randomly drawn in the movie and kept if there is no ROI in it.

Data augmentation

Movie augmentation

The movies acquired at higher spatial and temporal resolution are much smaller compared to the majority of movies once rescaled in the reference scale. As such, this kind of data was under-represented in the training data, with only a few events per movie. To reduce this bias, we doubled these movies by downsampling temporally each original movie two times and introducing a shift in the frames extracted between the two repetitions. These movies are indicated with "_aug" at the end of the names in the available dataset.

Image sequences augmentation

We also performed data augmentation on the training cropped image sequences. To augment the generalisation of the training, the augmentation was done on each training window with the addition of small temporal and spatial shifts, gaussian noise, white/black squared, and illumination on all windows or one time-frame.

Data imbalance

The events that we classified (nothing, extrusion, division, SOP) do not have the same frequency in all movies. To reduce the possible imbalance between the representation of these events in the training data, we reduced the number of windows of each over-represented event used in the training to have a number of training data similar than the less represented events. This option can be turned off in the pipeline with the boolean parameter “balance”.

Data rescaling

To homogenise the training dataset and have events of similar duration/size, we rescaled all our dataset to the same temporal and spatial resolution of 5 min/frame and 0.275 µm/pixel. Rescaling of the movies and ROIs was done with a Fiji macro. DeXNet networks were trained with input windows of this reference scale. A division event was visible on 2 to 3 time-frames, cell extrusion on 4 to 5 frames, and a cell had a typical diameter of 25 pixels. Therefore, the DeXtrusion pipeline must rescale all the movies to this reference scale before applying the classification. Since characteristic event durations and cell sizes can vary between tissues/organisms, we used these “cellular” features for rescaling rather than absolute times or distances.

Data to test generalisation

We tested the robustness of DeXtrusion to different unseen datasets. The first dataset was composed of 5 movies of *Drosophila* pupal notum depleted for EGFR (UAS-EGFR-dsRNA) (Valon et al., 2021). We do not have manual annotations on these movies, so they were not used in the training data. However, 3 movies in the original dataset were acquired with the same imaging and genetic conditions. This dataset was thus considered very similar to the training data.

The second dataset was a large movie of the *Drosophila* pupal wing from (Etournay et al., 2015) (3879x1947 pixels, 200 time-frames). This tissue was not represented in the training data, but the organisation of the epithelium and cell shape is very similar to the pupal notum. This dataset was considered similar to training data. We manually annotated a few extrusions (59) spanning the movie to evaluate the recall of DeXtrusion in this sample. We extracted a small part of the movie (495x444 pixels and 200 frames) and annotated this cropped movie to use as retraining data with 28 extrusions, 29 divisions and 19 additional controls for reinforcement. It is important to point out that the retraining window being inside the tested movie does not allow to

assess the performance of the retraining on fully independent data. Moreover, two cell extrusions of the 28 used for retraining were also present in the 59 used for testing, thus slightly biasing the performance assessment. Note that the field of view of the movie contains the whole wing but also external tissue on the top and bottom parts. We focused the quantification only on the ROIs that were fully inside the wing.

The third dataset was composed of four movies of larval epithelial cells (LECs) of the *Drosophila* pupal abdominal (Davis et al., 2022). The annotation of cell extrusions in the larval epidermal cells was obtained using the segmentation mask provided in the original dataset (Tapon and Salbreux, 2022) through tissue miner (Etournay et al., 2016). We focused our test only on larval cells and excluded histoblasts (the nest of small cells) since there is hardly any extrusion in this population and decided to focus on cells different from the pupal notum. We estimated the typical cell diameter of the larval cells to 80 pixels and the extrusion duration to 10 time-frames and used these values to rescale the movies for the pipeline. To estimate the Recall, we compared DeXtrusion outputs with the generated extrusion annotations on the original movie. However, since cell size and extrusion durations were much higher in these movies, we adjusted the thresholds of spatial and temporal distances to count matching detection (50 pixels xy distance, ~half a cell, and 8 time-frames). Note that the scores were always calculated on the 4 movies, even when some were used in the retraining data which could induce a slight positive bias. However, the 2 movies used for retraining were the ones on which DeXtrusion results were best even before retraining, and the most drastic improvements came from the 2 other movies.

DeXtrusion source code

DeXtrusion is available open-source on gitlab at https://gitlab.pasteur.fr/gletort/dextrusion under the BSD-3 license. The trained neural networks, Jupyter notebooks and Fiji macros to use DeXtrusion are available and described in this repository. DeXtrusion main code is deployed as a python module that can be installed through the pip installer package: https://pypi.org/project/dextrusion/. Instructions to install and use DeXtrusion are given in our gitlab repository.

To ease its usage, we proposed Jupyter notebooks that are dedicated to tasks such as network training, retraining or DeXtrusion detection on new movies. To visualize the

results as probability maps or ROIs, Fiji macros are also available. To handle input/output between our python code and Fiji, we used two specific python modules in our pipeline: “roifile” (Gohlke, 2022a) and “tifffile” (Gohlke, 2022b).

DeXNet architecture and training

Initial training

To categorize sliding windows by taking into account temporal information, we based the architecture of our neural networks on the Gated Recurrent Unit (GRU) architecture (Cho et al., 2014). We tested different variations of the architecture and hyperparameters (e.g.: number of layers, number of features by layer, number of epochs, size of the sliding windows). The final architecture is represented in **Fig. 1C** and the full detailed architecture can be found in the source code in the Network.py file. The parameters used for training one network are summarised in the configuration file associated with each DeXNet. For training the neural network to categorise the window as containing an event or not, we used the categorical cross-entropy loss. The number of event classified by DeXNet (control/extrusion, or control/extrusion/SOP or control/extrusion/SOP/division) was managed in the network architecture simply by fixing the size of the final output vector (**Fig. 1C**, last layer). This final layer is the result of a “softmax” function, so its values (2, 3 or 4 values) will add up to 1, like probabilities of event.

In our Gitlab repository, we proposed 2 DeXNets trained for controls and extrusions classifications (notum_Ext, used for **Fig. 1** and **Fig. 2**), controls, extrusions and SOPs classifications (notum_ExtSOP, **Fig. 3**), controls, extrusions, SOPs and cell divisions classifications (notum_EXTSOPDiv, **Fig. 3**), and trained on all events and all data (train and test) pooled together (notum_all, used for **Fig. 4**).

Optimisation and retraining

The number of categories within the model can be easily tuned by the user and only affects the last layer (densely connected layer) leading to the final prediction. As a result, it is easy to add new classes of cellular events to train or retrain the model and do predictions of these events without affecting the overall architecture of the model. We used that strategy in order to add the prediction of SOPs and division which were often wrongly labelled as extrusion (false positives) by the pipeline. We then extended that idea by manually selecting events classified as extrusions that were in fact “no

events" in the training data. We selected typical false positive events (e.g. transient cell constriction, loss of focus at the border of the tissue) and forced their classification as "no events" by adding them to the training dataset. Finally, the inherent stochasticity of the training (time sequences generated, weight initialisation, stochastic gradient descent) generates a diversity of final models even when trained on the same dataset. We used that opportunity in order to further optimise the pipeline by training different models on the same dataset. The two best ones were then combined in the pipeline making independent predictions on the input movies. The output probability maps are then simply averaged which increases the robustness of the model.

Evaluation of DeXtrusion results

DeXNet evaluation

The performance of DeXNet networks was measured by the accuracy of the results during training, and with the confusion matrix of the classifications on the test dataset after training (**Fig.1E**).

Pipeline evaluation

Comparison with manual annotations

To estimate the quality of DeXtrusion detections, we computed the accuracy ((TP+TN)/(TP+TN+FP+FN)), the precision (TP/(TP+FP)) and recall (TP/(TP+FN)) of the resulting ROIs compared to manual annotated ROIs (TP: True Positive, TN: True Negative, FP: False positive, FN: False Negative). ROIs were considered to be the same (between results and manual annotations) if they were within a spatial distance of 15 pixels and a temporal distance of 4 frames (in the reference scale). The Jupyter notebook dextrusion_CompareRois.ipynb allows us to calculate these scores and to choose the threshold distances to consider ROIs as the same. To consider both precision and recall at once, we also measured the F1-score (TP/(TP+(FP+FN)/2)) to evaluate the performance of our pipeline.

Measure of False Positive without manual annotations

For movies on which we do not have manual annotations, we cannot measure the recall as this would necessitate full annotations of all the events. We estimated the percentage of False Positive detections by selecting randomly a high number of resulting ROIs and examining each ROI manually to decide if the hit was correct or

not. The Fiji macro deXtrusion_scoreROIs_Random.ijm allows to do it and gives the resulting percentage.

DeXNet Optimisation

Once the model architecture was fixed, we optimised most of the model parameters to achieve the best possible results on the prediction of extrusion. For this, we started with the optimisation of the exploration time window (**Fig. S1 A-D**) as it is the first input of the model. We tested different size of the time window and kept a size of 10 (a good compromise between best score and calculation time). The training gave the best results when the exploration window was temporally centered on extrusion (5 frames before, 5 frames after) (**Fig. S1 E-G**). We then used this parameter to assess the effect of the window's xy size. The best results were obtained for a half size of 21px. While this parameter does not yield the best precision (**Fig. S1H**), it gives the best recall out of all parameter values explored (**Fig. S1I**). Recall was the metric we tried to optimise the most to avoid missing any extrusion. Moreover, the prediction time increased linearly with the window size (**Fig. S1J**). As a result, once pondered by time (precision*recall/prediction time), a size of 21px appears as the best value (**Fig. S1K**).

We then explored the impact of the different model thresholds on the model results (**Fig. S2**). We first fixed a volume threshold of 800px to assess the impact of the probability threshold (**Fig. S2A-C**). While the best f1-score is obtained for a probability threshold of 200, it comes at the expanse of a lower Recall (**Fig. S2A & C**). Thus, we set up to use a value of 180 for that parameter (second best but higher Recall, **Fig. S2A**) and then explore the impact of the volume threshold following similar reasoning, which led us to pick a value of 800 for that parameter.

Finally, we tried to optimise the hyperparameters (parameters of the model for training) (**Fig. S3**). First, we explore hyperparameters during the training of a model predicting only 2 classes (extrusions vs nothing). After parameters exploration we selected an augmentation of 3 (**Fig. S3A**), a number of epochs of 40 (**Fig. S3B**), a number of initial CNN filters of 8 (**Fig. S3C**) and a number of reinforcements of 5 (**Fig. S3D**). We then optimised the model by adding new predicting classes (SOPs and division, **Fig. 2**) and thus applied the same approach to the model prediction with 4 classes (extrusions, SOPs, divisions, nothing). Changing the number of epochs had a very limited impact

and we therefore kept a number of 40 epochs for the final model and an augmentation of 2.

All selected parameters are summarized in **Supplementary table 3.**

Acknowledgements

We would like to acknowledge the Image Analysis Hub of the Institut Pasteur for discussions and advice for this work. We would like to thank Jean-Yves Tinevez, Kevin Yamauchi and Raphaël Etournay for critical reading of the manuscript. We would also like to thank Raphaël Etournay for sharing the pupal wing movie and Nic Tapon for providing the original and segmentation of the pupal abdomen movies. We acknowledge the help of the HPC Core Facility of the Institut Pasteur for this work.

Authors contributions

AV and GL contributed equally to this work. AV and LV initiated the project and contributed to the dataset. AV gathered the full dataset, designed the model and conducted initial testing. GL revised the model and the code and extended it for the prediction on full movies. Both AV and GL contributed equally to the figure and results. AV, GL and RL designed the project and wrote the manuscript.

Competing interests

The authors declare no competing interests.

Funding

AV was supported by a PhD grant from the doctoral school “Complexité du Vivant” Sorbonne Université and from an extension grant of La Ligue contre le Cancer, work in RL lab is supported by the Institut Pasteur (G5 starting package), the ERC starting grant CoSpaDD (Competition for Space in Development and Disease, grant number 758457), the ANR CoECECa, and the CNRS (UMR 3738). We would also like to thank Danielle Casteran and her family for their generous donation which helped to finalise this project.

Data availability

DeXtrusion is distributed as a Python module, available open source on GitLab (https://gitlab.pasteur.fr/gletort/dextrusion) along with our trained neural networks and

scripts (Jupyter notebooks and Fiji macros) to facilitate its usage. Our annotated datasets used for training of the pipeline are fully available on Zenodo (Villars et al., 2023) (https://doi.org/10.5281/zenodo.7586394).

References

Aigouy, B., Cortes, C., Liu, S., Prud'Homme, B., 2020. EPySeg: a coding-free solution for automated segmentation of epithelia using deep learning. Development 147.

Aigouy, B., Farhadifar, R., Staple, D.B., Sagner, A., Roper, J.C., Julicher, F., Eaton, S., 2010. Cell flow reorients the axis of planar polarity in the wing epithelium of Drosophila. Cell 142, 773-786.

Aikin, T.J., Peterson, A.F., Pokrass, M.J., Clark, H.R., Regot, S., 2020. MAPK activity dynamics regulate non-cell autonomous effects of oncogene expression. Elife 9.

Ambrosini, A., Gracia, M., Proag, A., Rayer, M., Monier, B., Suzanne, M., 2017. Apoptotic forces in tissue morphogenesis. Mech Dev 144, 33-42.

Aspert, T., Hentsch, D., Charvin, G., 2022. DetecDiv, a generalist deep-learning platform for automated cell division tracking and survival analysis. Elife 11.

Bock, F.J., Sedov, E., Koren, E., Koessinger, A.L., Cloix, C., Zerbst, D., Athineos, D., Anand, J., Campbell, K.J., Blyth, K., Fuchs, Y., Tait, S.W.G., 2021. Apoptotic stress-induced FGF signalling promotes non-cell autonomous resistance to cell death. Nat Commun 12, 6572.

Cho, K., van Merriënboer, B., Bahdanau, D., Bengio, Y., 2014. On the Properties of Neural Machine Translation: Encoder–Decoder Approaches. Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation.

Davis, J.R., Ainslie, A.P., Williamson, J.J., Ferreira, A., Torres-Sanchez, A., Hoppe, A., Mangione, F., Smith, M.B., Martin-Blanco, E., Salbreux, G., Tapon, N., 2022. ECM degradation in the Drosophila abdominal epidermis initiates tissue growth that ceases with rapid cell-cycle exit. Curr Biol 32, 1285-1300 e1284.

Etournay, R., Merkel, M., Popovic, M., Brandl, H., Dye, N.A., Aigouy, B., Salbreux, G., Eaton, S., Julicher, F., 2016. TissueMiner: A multiscale analysis toolkit to quantify how cellular processes create tissue dynamics. Elife 5.

Etournay, R., Popovic, M., Merkel, M., Nandi, A., Blasse, C., Aigouy, B., Brandl, H., Myers, G., Salbreux, G., Julicher, F., Eaton, S., 2015. Interplay of cell dynamics and epithelial tension during morphogenesis of the Drosophila pupal wing. Elife 4, e07090.

Fan, Y., Bergmann, A., 2008. Apoptosis-induced compensatory proliferation. The Cell is dead. Long live the Cell! Trends Cell Biol 18, 467-473.

Farhadifar, R., Roper, J.C., Aigouy, B., Eaton, S., Julicher, F., 2007. The influence of cell mechanics, cell-cell interactions, and proliferation on epithelial packing. Curr Biol 17, 2095-2104.

Gagliardi, P.A., Dobrzynski, M., Jacques, M.A., Dessauges, C., Ender, P., Blum, Y., Hughes, R.M., Cohen, A.R., Pertz, O., 2021. Collective ERK/Akt activity waves orchestrate epithelial homeostasis by driving apoptosis-induced survival. Dev Cell 56, 1712-1726 e1716.

Gallusser, B., Stieber, M., VWeigert, M., 2023. Self-supervised dense representation learning for live-cell microscopy with time arrow prediction. arXiv. https://doi.org/10.48550/arXiv.2305.05511

Gho, M., Bellaiche, Y., Schweisguth, F., 1999. Revisiting the Drosophila microchaete lineage: a novel intrinsically asymmetric cell division generates a glial cell. Development 126, 3573-3584.

Gohlke, C., 2022a. roifile: v2022.9.19. https://zenodo.org/record/7094778#.Y-Ik-HbMJdg.

Gohlke, C., 2022b. tifffile: v2022.5.4. https://zenodo.org/record/6795861#.Y-IkNXbMJdg.

Guirao, B., Bellaiche, Y., 2017. Biomechanics of cell rearrangements in Drosophila. Curr Opin Cell Biol 48, 113-124.

Guirao, B., Rigaud, S.U., Bosveld, F., Bailles, A., Lopez-Gay, J., Ishihara, S., Sugimura, K., Graner, F., Bellaiche, Y., 2015. Unified quantitative characterization of epithelial tissue development. Elife 4.

Hallou, A., Yevick, H.G., Dumitrascu, B., Uhlmann, V., 2021. Deep learning for bioimage analysis in developmental biology. Development 148.
Herbert, S., Valon, L., Mancini, L., Dray, N., Caldarelli, P., Gros, J., Esposito, E., Shorte, S.L., Bally-Cuif, L., Aulner, N., Levayer, R., Tinevez, J.Y., 2021. LocalZProjector and DeProj: a toolbox for local 2D projection and accurate morphometrics of large 3D microscopy images. BMC Biol 19, 136.
Hoshika, S., Sun, X., Kuranaga, E., Umetsu, D., 2020. Reduction of endocytic activity accelerates cell elimination during tissue remodeling of the Drosophila epidermal epithelium. Development 147.
Huang, X., Li, Z., Zhang, M., Gao, S., 2022. Fusing hand-crafted and deep-learning features in a convolutional neural network model to identify prostate cancer in pathology images. Front Oncol 12, 994950.
Kabir, M.A., Kharel, A., Malla, S., Kreis, Z.J., Nath, P., Wolfe, J.N., Hassan, M., Kaur, D., Sari-Sarraf, H., Tiwari, A.K., Ray, A., 2022. Automated detection of apoptotic versus nonapoptotic cell death using label-free computational microscopy. J Biophotonics 15, e202100310.
Kawaue, T., Yow, I., Le, A.P., Lou, Y., Loberas, M., Shagirov, M., Prost, J., Hiraiwa, T., Ladoux, B., Toyama, Y., 2021. Mechanics defines the spatial pattern of compensatory proliferation. bioRxiv, 2021.2007.2004.451019.
La Greca, A.D., Perez, N., Castaneda, S., Milone, P.M., Scarafia, M.A., Mobbs, A.M., Waisman, A., Moro, L.N., Sevlever, G.E., Luzzani, C.D., Miriuka, S.G., 2021. celldeath: A tool for detection of cell death in transmitted light microscopy images by deep learning-based visual recognition. PLoS One 16, e0253666.
LeCun, Y., Bengio, Y., Hinton, G., 2015. Deep learning. Nature 521, 436-444.
Levayer, R., Dupont, C., Moreno, E., 2016. Tissue Crowding Induces Caspase-Dependent Competition for Space. Curr Biol 26, 670-677.
Mahecic, D., Stepp, W.L., Zhang, C., Griffie, J., Weigert, M., Manley, S., 2022. Event-driven acquisition for content-enriched microscopy. Nat Methods 19, 1262-1267.
Marinari, E., Mehonic, A., Curran, S., Gale, J., Duke, T., Baum, B., 2012. Live-cell delamination counterbalances epithelial growth to limit tissue overcrowding. Nature 484, 542-545.
Matamoro-Vidal, A., Cumming, T., Davidovic, A., Levayer, R., 2022. Patterned apoptosis modulates local growth and tissue shape in a fast-growing epithelium. bioRxiv, 2022.2003.2011.484029.
Mesa, K.R., Kawaguchi, K., Cockburn, K., Gonzalez, D., Boucher, J., Xin, T., Klein, A.M., Greco, V., 2018. Homeostatic Epidermal Stem Cell Self-Renewal Is Driven by Local Differentiation. Cell Stem Cell 23, 677-686 e674.
Michel, M., Dahmann, C., 2020. Tissue mechanical properties modulate cell extrusion in the Drosophila abdominal epidermis. Development 147.
Mockl, L., Roy, A.R., Moerner, W.E., 2020. Deep learning in single-molecule microscopy: fundamentals, caveats, and recent developments [Invited]. Biomed Opt Express 11, 1633-1661.
Monier, B., Gettings, M., Gay, G., Mangeat, T., Schott, S., Guarner, A., Suzanne, M., 2015. Apico-basal forces exerted by apoptotic cells drive epithelium folding. Nature 518, 245-248.
Moreno, E., Valon, L., Levillayer, F., Levayer, R., 2019. Competition for Space Induces Cell Elimination through Compaction-Driven ERK Downregulation. Curr Biol 29, 23-34 e28.
Perez-Garijo, A., Fuchs, Y., Steller, H., 2013. Apoptotic cells can induce non-autonomous apoptosis through the TNF pathway. Elife 2, e01004.
Phan, H.T.H., Kumar, A., Feng, D., Fulham, M., Kim, J., 2019. Unsupervised Two-Path Neural Network for Cell Event Detection and Classification Using Spatiotemporal Patterns. IEEE Trans Med Imaging 38, 1477-1487.
Ranft, J., Basan, M., Elgeti, J., Joanny, J.F., Prost, J., Julicher, F., 2010. Fluidization of tissues by cell division and apoptosis. Proc Natl Acad Sci U S A 107, 20863-20868.
Roellig, D., Theis, S., Proag, A., Allio, G., Benazeraf, B., Gros, J., Suzanne, M., 2022. Force-generating apoptotic cells orchestrate avian neural tube bending. Dev Cell.
Rosenblatt, J., Raff, M.C., Cramer, L.P., 2001. An epithelial cell destined for apoptosis signals its neighbors to extrude it by an actin- and myosin-dependent mechanism. Curr Biol 11, 1847-1857.

Segebarth, D., Griebel, M., Stein, N., von Collenberg, C.R., Martin, C., Fiedler, D., Comeras, L.B., Sah, A., Schoeffler, V., Luffe, T., Durr, A., Gupta, R., Sasi, M., Lillesaar, C., Lange, M.D., Tasan, R.O., Singewald, N., Pape, H.C., Flath, C.M., Blum, R., 2020. On the objectivity, reliability, and validity of deep learning enabled bioimage analyses. Elife 9.
Shkolyar, A., Gefen, A., Benayahu, D., Greenspan, H., 2015. Automatic detection of cell divisions (mitosis) in live-imaging microscopy images using Convolutional Neural Networks. Annu Int Conf IEEE Eng Med Biol Soc 2015, 743-746.
Suzanne, M., Petzoldt, A.G., Speder, P., Coutelis, J.B., Steller, H., Noselli, S., 2010. Coupling of apoptosis and L/R patterning controls stepwise organ looping. Curr Biol 20, 1773-1778.
Takeuchi, Y., Narumi, R., Akiyama, R., Vitiello, E., Shirai, T., Tanimura, N., Kuromiya, K., Ishikawa, S., Kajita, M., Tada, M., Haraoka, Y., Akieda, Y., Ishitani, T., Fujioka, Y., Ohba, Y., Yamada, S., Hosokawa, Y., Toyama, Y., Matsui, T., Fujita, Y., 2020. Calcium Wave Promotes Cell Extrusion. Curr Biol.
Tapon, N., Salbreux, G., 2022. Supplementary data for Davis et al Current Biol 2022. The Francis Crick Institute. Collection. https://doi.org/10.25418/crick.c.5787494.v1
Teng, X., Qin, L., Le Borgne, R., Toyama, Y., 2017. Remodeling of adhesion and modulation of mechanical tensile forces during apoptosis in Drosophila epithelium. Development 144, 95-105.
Tetley, R.J., Staddon, M.F., Heller, D., Hoppe, A., Banerjee, S., Mao, Y., 2019. Tissue Fluidity Promotes Epithelial Wound Healing. Nat Phys 15, 1195-1203.
Toyama, Y., Peralta, X.G., Wells, A.R., Kiehart, D.P., Edwards, G.S., 2008. Apoptotic force and tissue dynamics during Drosophila embryogenesis. Science 321, 1683-1686.
Valon, L., Davidović, A., Levillayer, F., Villars, A., Chouly, M., Cerqueira-Campos, F., Levayer, R., 2021. Robustness of epithelial sealing is an emerging property of local ERK feedback driven by cell elimination. Developmental Cell 56, 1700-1700.
Villars, A., Letort, G., Valon, L., Levayer, R., 2023. Annotated dataset DeXtrusion. https://doi.org/10.5281/zenodo.7586394.
Villars, A., Levayer, R., 2022. Collective effects in epithelial cell death and cell extrusion. Curr Opin Genet Dev 72, 8-14.
Villars, A., Matamoro-Vidal, A., Levillayer, F., Levayer, R., 2022. Microtubule disassembly by caspases is an important rate-limiting step of cell extrusion. Nat Commun 13, 3632.
Yamamoto, T., Cockburn, K., Greco, V., Kawaguchi, K., 2022. Probing the rules of cell coordination in live tissues by interpretable machine learning based on graph neural networks. PLoS Comput Biol 18, e1010477.

Figure legends

Figure 1: DeXNet, a neural network to recognise cell extrusions in image sequences in epithelia.

A: Snapshot of a developing *Drosophila* pupal notum marked with E-cad::GFP (Knock-in) with overlaying extrusions manually detected over 16h of development. Histograms represent the distribution of cell extrusion along the anteroposterior axis (bottom histogram) and along the left-right axis (right histogram) for a total of 720 extrusions. Scale bar=50 µm.

B: Representative image sequence centred on an extrusion event (top row, blue arrow points at the end of extrusion), or representing a control sequence with no event (bottom row). The time is given in frame (here, one frame=5 minutes). Scale bar=5 µm.

C-C': Schematic of the model architecture called DeXNet. **C.** Main model. The image sequence composed of 10 time-frames (T) of 45 by 45 pixels (px) is passed first to a CNN detailed in **C'** which encodes each image of the sequence in a vector of 64 features. The resulting matrix for the sequence (10Tx64, representing 10 time-points) is itself fed into a Gated Recurrent Unit (GRU) to consider temporal information. This is then passed to a dropout normalisation layer before going through a sequence of densely connected layers (dense). Finally, it predicts with a probability whether the input sequence is an extrusion or not. **C':** Detailed architecture of the encoding CNN. Each image in the sequence goes into a sequence of convolution (Con2D), Batch normalisation and max pooling (MaxPool, all this is repeated 4 times). The convolution was made using n=8 filters which are multiplied by 2 at each layer thereby increasing the first dimension (8,16,32,64) while the max pooling reduces the other dimension (45,22,11,5). Finally, this output (64x5x5) goes through a final step of max pooling encoding the image in a final vector of dimension (1x64).

D: loss (left y-axis) and accuracy (right y-axis) curves for training data (bold lines) and validation data (dotted lines). The loss function is an estimation of the distance between the prediction and the real data (equivalent to an error estimation). Accuracy is defined by the proportion of correct predictions out of all the predictions (see **Material and Methods**).

E: Confusion matrix showing the accuracy of the DeXtrusion model. Orange-coloured boxes show the number of correctly predicted events (light orange are True Negatives, darker orange shows True Positives). Grey events are the number of Wrongly predicted events (light grey are False Negatives and darker grey show False Positives).

F: Representative image sequences showing example events for each cateogry in **E** (TN: True Negatives – Events correctly classified as controls, TP: True Positives – Extrusions events correctly classified as extrusions, FN: False Negatives – Extrusions wrongly classified as control events, FP: False Positives – Control events wrongly classified as extrusions). Scale bar=5 µm.

Figure 2: DeXtrusion pipeline to detect extrusions on full movies.

A: Schematic representation of the DeXtrusion pipeline. The input movie is first rescaled to the reference scale. Then, cropped image sequences are extracted, spanning the entire movie with an overlap both spatially and temporally. Each image sequence is then processed through our neural network DeXNet. The resulting probability of the presence of an extrusion is added around the central position of the image sequence to build a probability map on the entire movie, before rescaling it to the original movie size. Single point events can also be generated by taking the centroids of high probability volumes in the probability map and exported as a Fiji ROI file.

B: Projection of a resulting probability map. Snapshot of the input movie, an epithelium labelled with E-cadherin-GFP (left) with the corresponding probability map (right). Probabilities are drawn as a color map, with values converted to 0 (black) - 255 (white) scale for visualisation. Scale bar=50 μm.

C,C': Example of a correctly detected extrusion (**C**) and a False Positive detection (**C'**, here a Sensory Organ Precursos, SOP). Image sequences of E-cadherin::GFP (top) from the original movie cropped around a correctly (**C**) or wrongly (**C'**) detected extrusion event and corresponding extrusion probability (bottom). Probabilities are colour coded from 0 (black) to 255 (white). White arrows indicate detected events. Scale bar=5 μm.

D,D': 2D histogram of the spatio-temporal distances between manually annotated extrusions and DeXtrusion results on the 7 test movies. The colour code represents the number of extrusions detected within a given temporal distance (x-axis, in time frames, 1 frame=5 minutes) and a given spatial distance (y-axis, in pixels, 1 pixel=0.275μm). The red dotted rectangle represents the events that are considered as matching with a manually annotated event (below the spatial and temporal threshold, see **Material and Methods**). The histogram of distribution for spatial distance is shown on the right, and the one for temporal distance is on the top. The thresholds are represented with red dotted lines. **D':** close-up view of the histograms of spatial and temporal distances to the area that contains the most observations (see grey square in **D**).

E: Cumulative number of extrusions during developmental time for DeXtrusion results (red) and manually annotated one (blue). Extrusions were detected on one test movie of 942*942 pixels and 200 frames.

F: Average absolute difference of the number of extrusions over time between DeXtrusion detections and manually annotated extrusions. From all test set movies (binned every 10 min from the start of the movie).

Figure 3: Optimisation of the model for the detection of extrusions

A-D: Image sequence events predicted as extrusion correctly (**A**) or not (**B-D**) by the initial DeXnet trained on 2 classes only (extrusion or no event). Top rows of each panel represent the image sequence, bottom rows represent the probability maps obtained by DeXtrusion. Scale bar= 5 µm. **A:** Image sequence representing an extrusion and its associated probability map. Yellow arrows point at the apical closure of the extruding cell. **B:** Image sequence representing a forming Sensory Organ Precursor (SOP) wrongly predicted as an extrusion and its associated probability map. Yellow arrows point at the small cell of the SOP which constricts and leads to misclassification. **C:** Image sequence representing a dividing cell wrongly predicted as an extrusion and its associated probability map. Yellow arrows point at the furrow formation during cytokinesis. **D:** Image sequence representing a control event wrongly predicted as an extrusion and its associated probability map.

E-E'': Changes in the model to optimise its prediction scores on extrusion. **E:** f1-score. **E':** Recall, **E'':** Precision, with the initial two-class model (ext: extrusions and no event), the inclusion of SOPs (ext, SOP: extrusions, SOPs, and no event), the inclusion of SOPs and cell divisions (ext, SOP, div: extrusions, SOPs, cell division and no event), including the 3 cellular events and reinforcement (reinforcement, see below), and using two independent networks (2net, see below). The results are shown for the best-trained network for each class model (see **Fig. S2 G-I** for the averaged).

F-F': Schematics of the steps added to the 4 classes model to increase its f1-score on predicting extrusions. **F:** Reinforcement consists in taking regions covering no event misclassified as extrusion, cropping them and adding them to the training set with a control label. This forces the model to learn that the previously misclassified events are in fact controls (no event). **F':** 2nets uses the two best independently trained

models (stochasticity results in models with slightly different weights and biases), and averages the probability map from these two models.

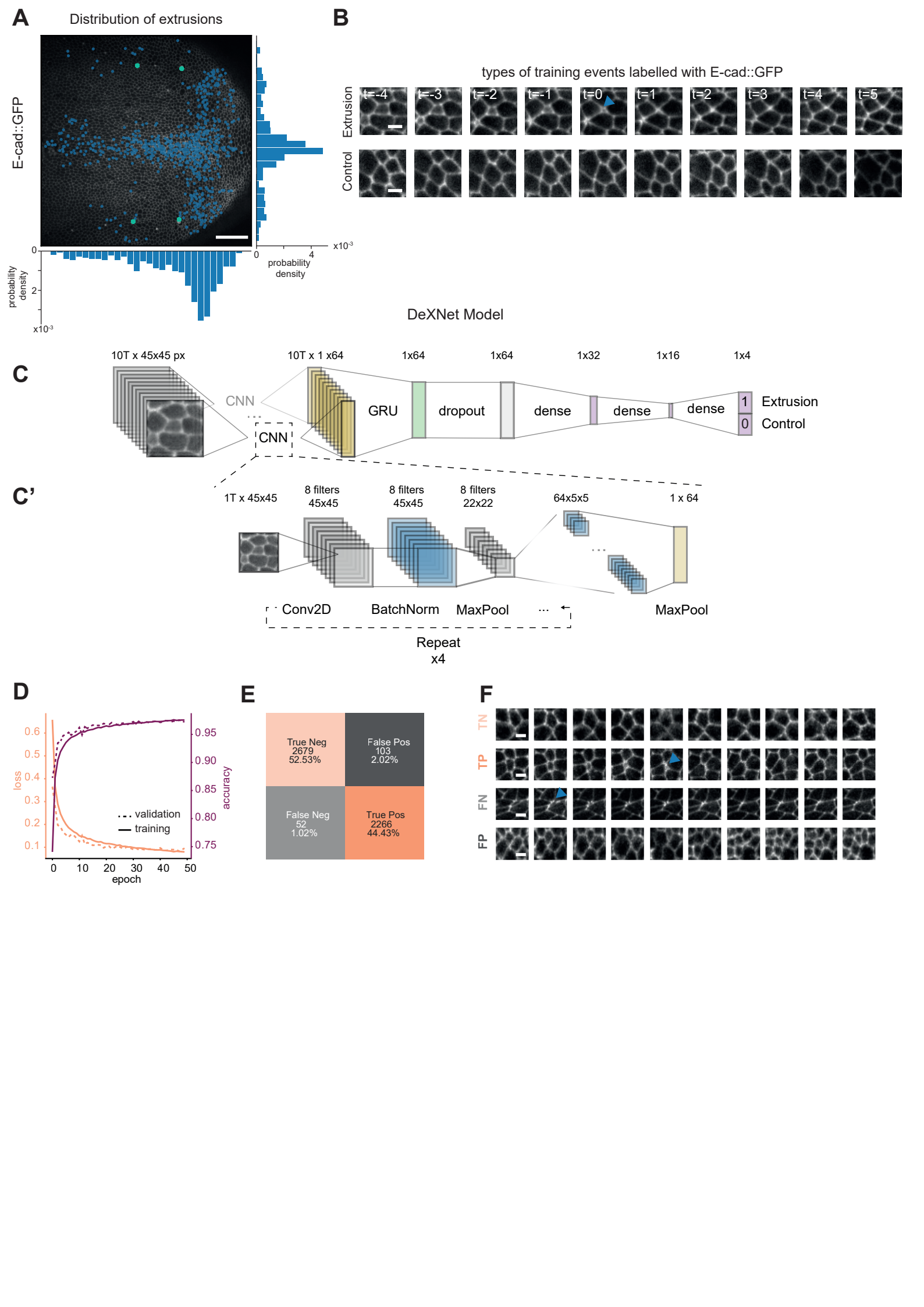
G: Image sequence showing the results of DeXtrusion predictions after optimisation, of extrusions (orange), cell divisions (pink) and differentiation (SOPs, blue) on the full-scale movie (pupal notum, local projection of E-cad::GFP) and overtime. Time is shown in hours After Pupal Formation (hAPF). Scale bar= 50 µm.

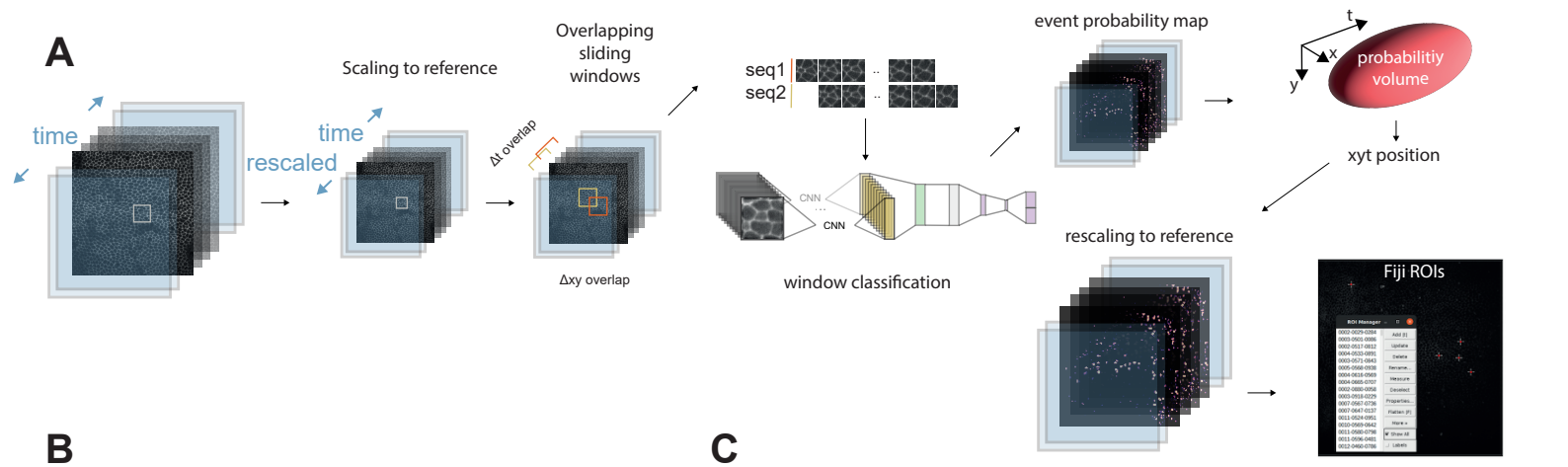
Figure 4: Generalisation of DeXtrusion to mutant context and other epithelia

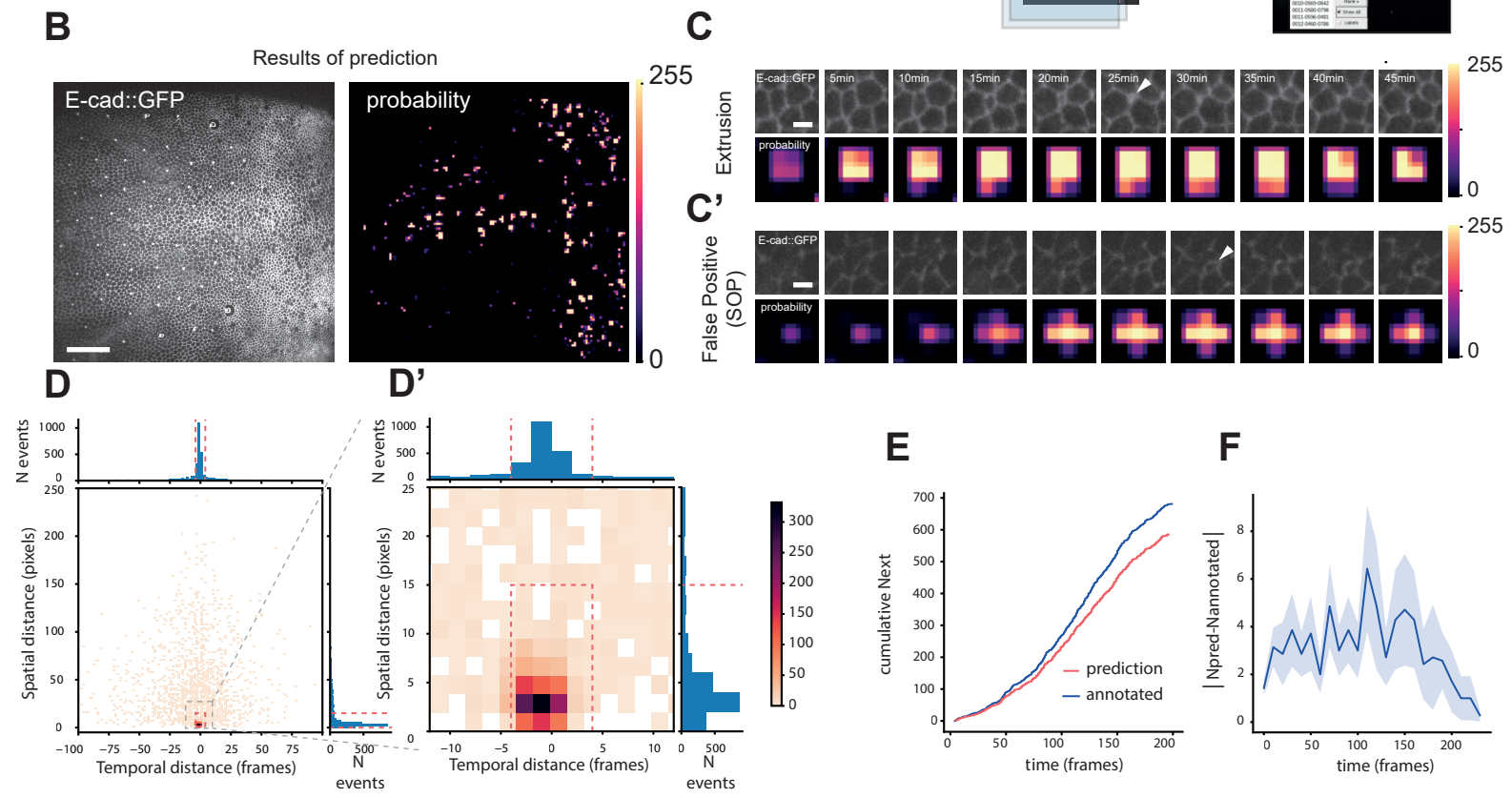
A: Schematic of a *Drosophila* pupa highlighting the different epithelia on which the model was tested. The model was trained on the notum (grey) and tested on the pupal wing (blue) and finally on the abdominal Larval Epithelial Cells (LECs, green).

B-D': Example image sequences of the different tissues used to test the generalisation of DeXtrusion and the corresponding extrusion phenotypes. **B:** Pupal notum epithelium expressing a RNAi against EGFR. Scale bar=50 µm. **B':** Example image sequence showing an extrusion cropped from **B**. Scale bar=5 µm **C:** Pupal wing epithelium from (Etournay et al., 2015). Scale bar=50 µm. **C':** Example image sequence showing an extrusion from **C.** Scale bar=5 µm. **D:** Pupal abdomen epithelium extracted from (Davis et al., 2022), results were computed only for LECs (bigger cells) excluding the histoblasts (smaller cells). Scale bar=50 µm. **D':** Example image sequence showing an extruding LEC from **D**. Scale bar=10 µm.

E: Manually computed precision for the prediction on the different tissues with increasing differences compared to the tissues for which DeXtrusion was trained on. Green shows the precision of DeXtrusion for notum expressing a RNAi against EGFR (example shown in **B-B'**). Light blue is the precision of DeXtrusion for the pupal wing epithelium (example shown in **C-C'**). Dark blue is the precision of DeXtrusion on the same tissue after retraining the model on a subset of extrusions. Light purple is the precision of DeXtrusion on LECs (example shown in **D-D'**). Purple is the prediction on the same tissue after retraining. Darker purple is the prediction on the same tissue after retraining and reinforcement.


A
Distribution of extrusions
E-cad::GFP
probability density
x10-3
B
types of training events labelled with E-cad::GFP
Extrusion
Control
t=-4
t=-3
t=-2
t=-1
t=0
t=1
t=2
t=3
t=4
t=5
DeXNet Model
C
10T x 45x45 px
10T x 1 x64
1x64
1x64
1x32
1x16
1x4
CNN
GRU
dropout
dense
dense
dense
1
0
Extrusion
Control
C'
1T x 45x45
8 filters 45x45
8 filters 45x45
8 filters 22x22
64x5x5
1 x 64
Conv2D
BatchNorm
MaxPool
MaxPool
Repeat x4
D
loss
accuracy
validation
training
epoch
E
True Neg 2679 52.53%
False Pos 103 2.02%
False Neg 52 1.02%
True Pos 2266 44.43%
F
TN
TP
FN
FP

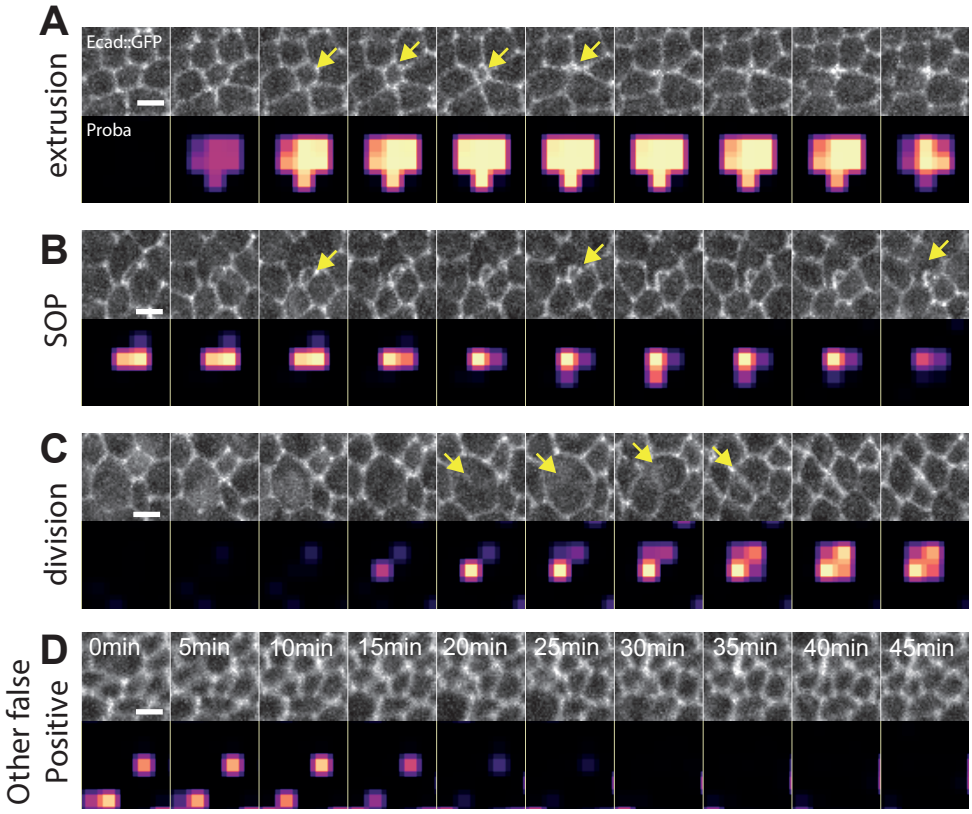

A
time
rescaled
Scaling to reference
Δt overlap
Overlapping sliding windows
Δxy overlap
seq1
seq2
CNN
CNN
window classification
event probability map
t
x
y
probabilitiy volume
xyt position
rescaling to reference
Fiji ROIs

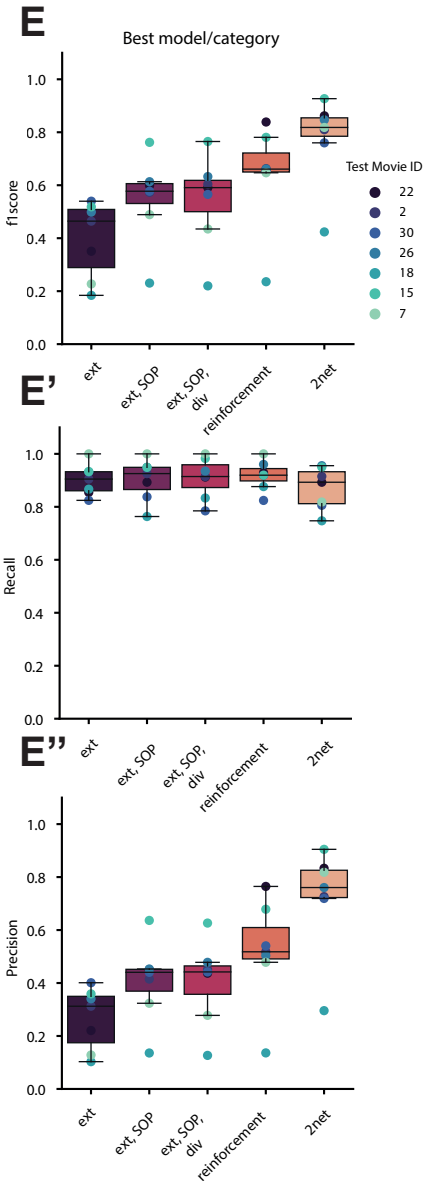

B
Results of prediction
E-cad::GFP
probability
255
0
C
Extrusion
C'
False Positive (SOP)
E-cad::GFP
probability
5min
10min
15min
20min
25min
30min
35min
40min
45min
255
0

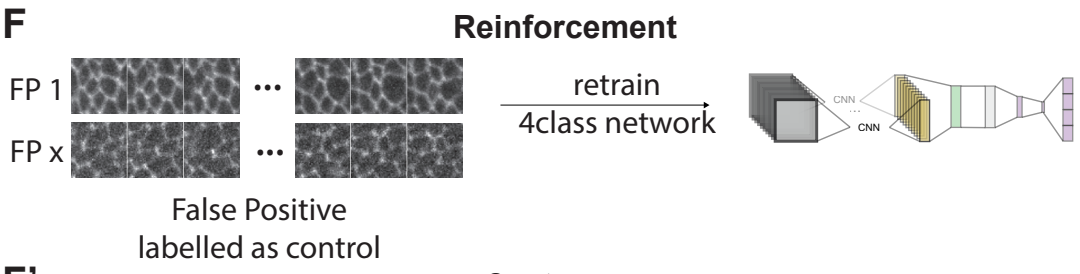
D
D'
N events
Spatial distance (pixels)
Temporal distance (frames)
N events
300
250
200
150
100
50
0

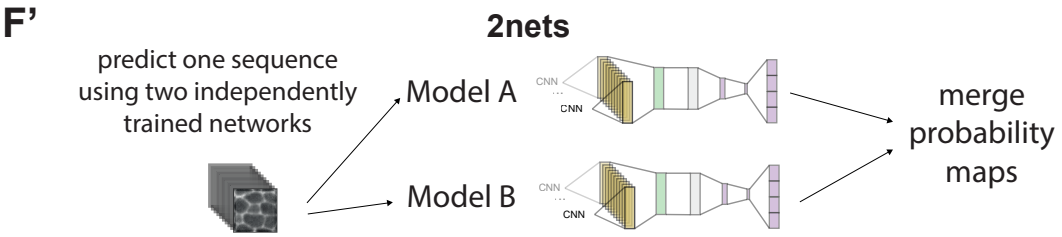
E
cumulative Next
prediction
annotated
time (frames)

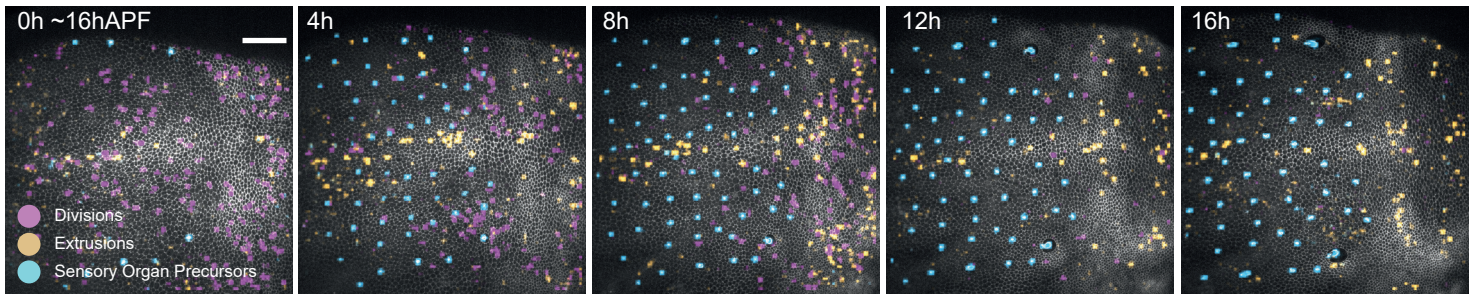
F
|Npred-Nannotated|
time (frames)

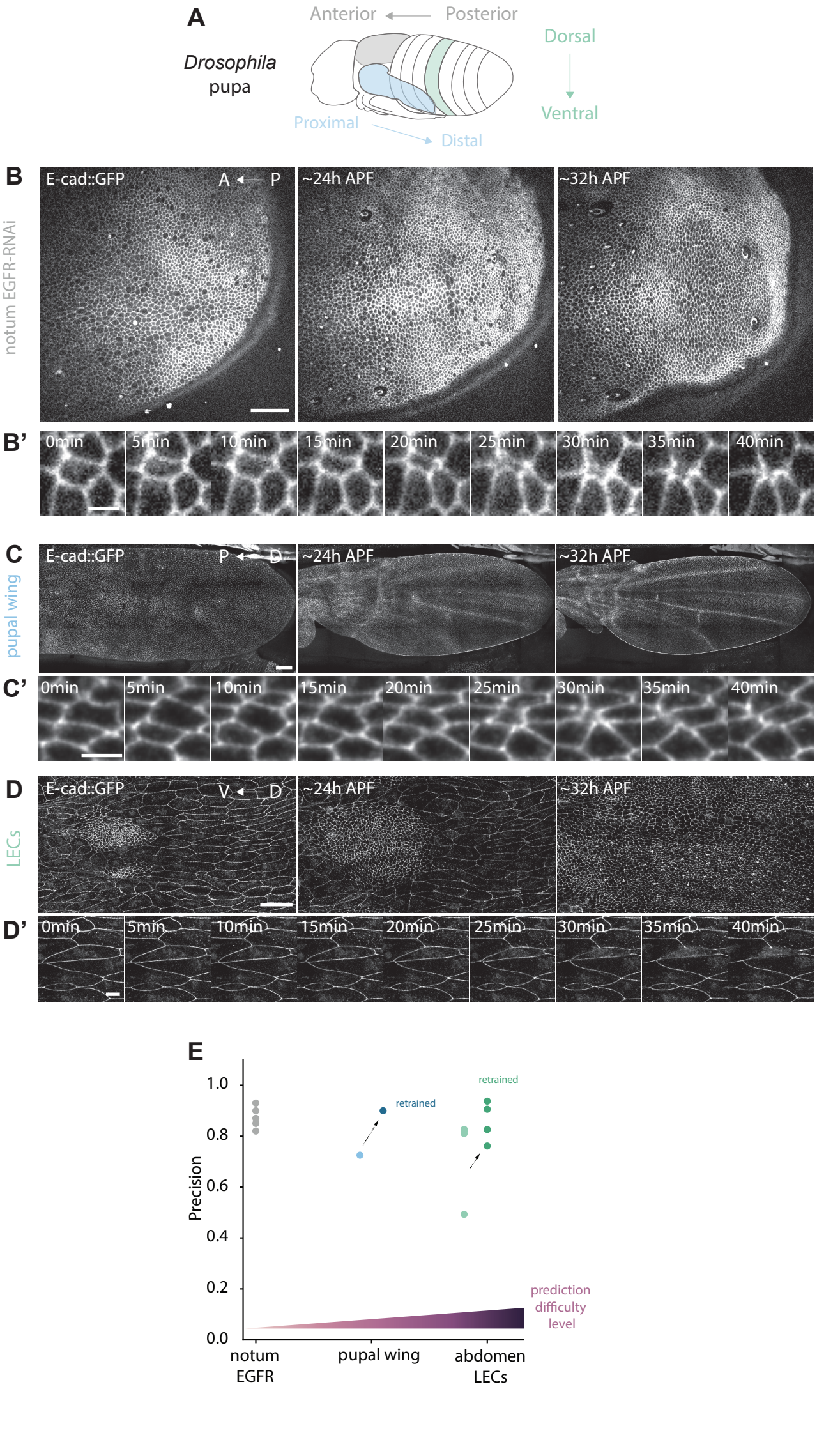

A
extrusion
Ecad::GFP
Proba
B
SOP
C
division
D
Other false Positive
0min
5min
10min
15min
20min
25min
30min
35min
40min
45min


E
Best model/category
f1score
Test Movie ID
22
2
30
26
18
15
7
ext
ext, SOP
ext, SOP, div
reinforcement
2net
E'
Recall
E''
Precision


F
Reinforcement
FP 1
FP x
False Positive
labelled as control
retrain
4class network
CNN


F'
2nets
predict one sequence
using two independently
trained networks
Model A
Model B
CNN
merge
probability
maps


G
0h ~16hAPF
4h
8h
12h
16h
Divisions
Extrusions
Sensory Organ Precursors


A
Drosophila
pupa
Anterior
Posterior
Dorsal
Ventral
Proximal
Distal
B
notum EGFR-RNAi
E-cad::GFP
A
P
~24h APF
~32h APF
B'
0min
5min
10min
15min
20min
25min
30min
35min
40min
C
pupal wing
E-cad::GFP
P
D
~24h APF
~32h APF
C'
0min
5min
10min
15min
20min
25min
30min
35min
40min
D
LECs
E-cad::GFP
V
D
~24h APF
~32h APF
D'
0min
5min
10min
15min
20min
25min
30min
35min
40min
E
Precision
1.0
0.8
0.6
0.4
0.2
0.0
retrained
retrained
prediction
difficulty
level
notum
EGFR
pupal wing
abdomen
LECs

Supplementary information

DeXtrusion: Automatic recognition of epithelial cell extrusion through machine learning *in vivo*

Alexis Villars[1,2*], Gaëlle Letort[1*], Léo Valon[1] and Romain Levayer[1§]

1. Department of Developmental and Stem Cell Biology, Institut Pasteur, Université de Paris Cité, CNRS UMR 3738, 25 rue du Dr. Roux, 75015 Paris, France

2. Current address: Department of Biosystems Science and Engineering, ETH Zürich, Mattenstrasse 26, 4058 Basel, Switzerland

* These authors contributed equally to this work

§ Lead contact, correspondance to: romain.levayer@pasteur.fr

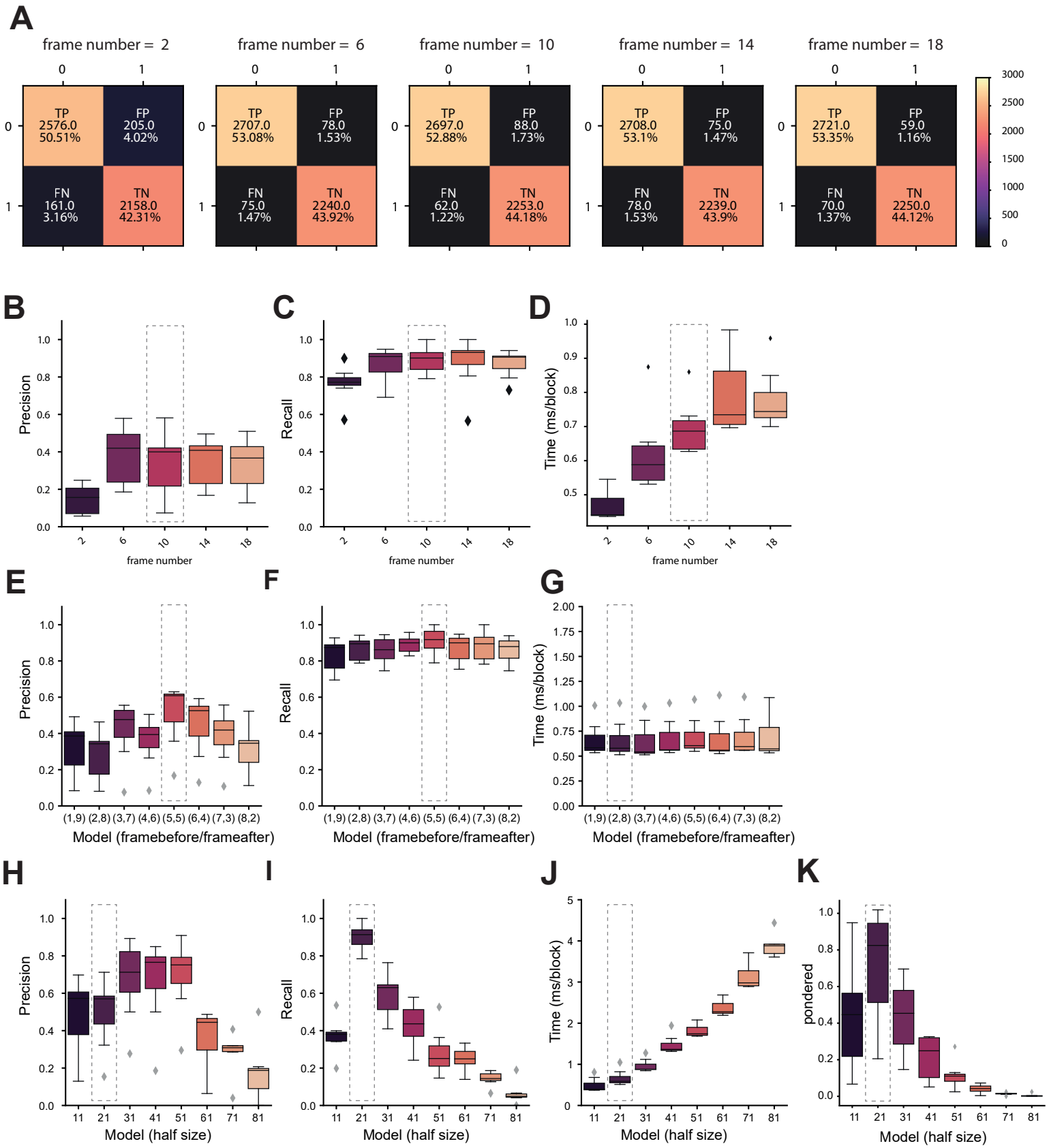

A
frame number = 2
TP 2576.0 50.51%
FP 205.0 4.02%
FN 161.0 3.16%
TN 2158.0 42.31%
frame number = 6
TP 2707.0 53.08%
FP 78.0 1.53%
FN 75.0 1.47%
TN 2240.0 43.92%
frame number = 10
TP 2697.0 52.88%
FP 88.0 1.73%
FN 62.0 1.22%
TN 2253.0 44.18%
frame number = 14
TP 2708.0 53.1%
FP 75.0 1.47%
FN 78.0 1.53%
TN 2239.0 43.9%
frame number = 18
TP 2721.0 53.35%
FP 59.0 1.16%
FN 70.0 1.37%
TN 2250.0 44.12%
B
Precision
frame number
C
Recall
frame number
D
Time (ms/block)
frame number
E
Precision
Model (framebefore/frameafter)
F
Recall
Model (framebefore/frameafter)
G
Time (ms/block)
Model (framebefore/frameafter)
H
Precision
Model (half size)
I
Recall
Model (half size)
J
Time (ms/block)
Model (half size)
K
pondered
Model (half size)

Figure S1: Optimisation of the search window size for extrusion detection

A: Confusion matrix showing the accuracy obtained for different temporal size of the search window (from 2 to 18 time frames). Orange-coloured boxes show the number of correctly predicted events (light orange are True Negatives, darker orange shows True Positives). Grey events are the number of Wrongly predicted events (light grey are False Negative and darker grey show False Positives). The total number of prediction as well as the proportion is given in each case. These values were obtained for a single round of network training for each condition.

B-D: Optimisation of the temporal size of the search window (from 2 to 18 time frames). For each duration, the Precision (**B**), Recall (**C**), and time of calculation (ms per block) (**D**), were estimated on the test dataset. Note that the trainings were not performed on the same set up as in **Figure S1 G**,**J** so the absolute time cannot be compared between these panels. Box plots show the median, the first and third quartile. Top and bottom bars are the maximal and minimal value. Diamonds are outliers.

E-G: Optimisation of the temporal positioning of the search window according to the termination of extrusion (manually detected, end of apical area closure), number of frames before or after the extrusion detection point. For each position, the Precision (**E**), Recall (**F**), and time of calculation (ms per block) (**G**), were estimated on the test dataset. The optimum was obtained for a search window centered on extrusion termination (5,5). Box plots show the median, the first and third quartile. Top and bottom bars are the maximal and minimal value. Diamonds are outliers.

H-K: Optimisation of the size (x y) of the square search window (half size in pixel after rescaling, 1 pixel=0.275 μm). For each position, the Precision (**H**), Recall (**I**), and time of calculation (ms per block) (**J**), were estimated on the training dataset. We computed then a ponderated parameter (**K**, see **Material and Methods**) which takes into account precision, recall and calculation time which peaks for 21px. Box plots show the median, the first and third quartile. Top and bottom bars are the maximal and minimal value. Diamonds are outliers.

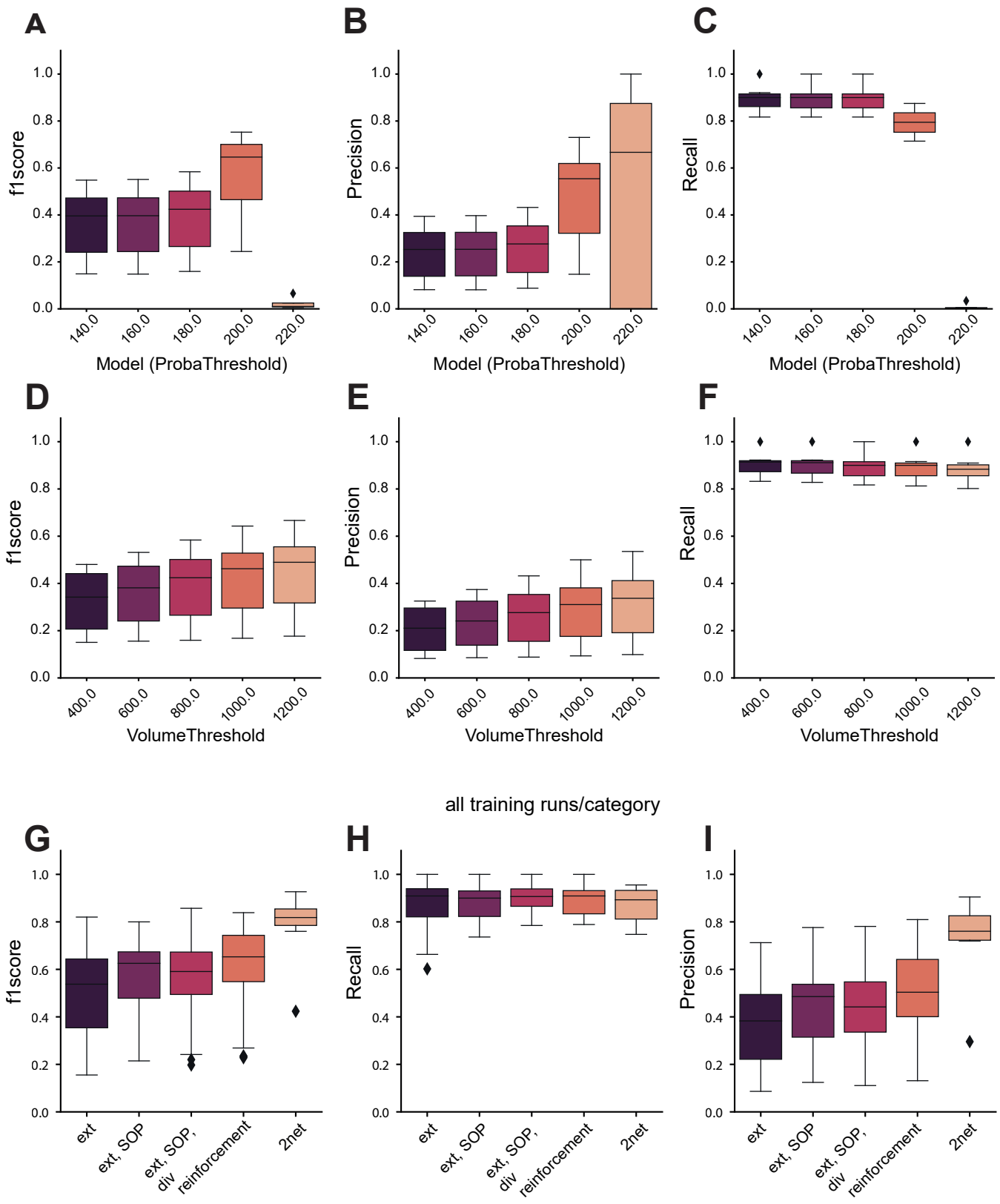

A
B
C
f1score
Precision
Recall
1.0
0.8
0.6
0.4
0.2
0.0
140.0
160.0
180.0
200.0
220.0
Model (ProbaThreshold)
D
E
F
400.0
600.0
800.0
1000.0
1200.0
VolumeThreshold
all training runs/category
G
H
I
ext
ext, SOP
ext, SOP, div
reinforcement
2net

Figure S2: Optimisation of probability thresholding for extrusion detection and detection of new categories of cellular events

A-C: Optimisation of the probability threshold used to assign an extrusion event. For each threshold, the f1 score (**A**), precision (**B**), and recall (**C**), were estimated on the test dataset. The optimum f1 score was obtained for 200 (a.u.), however we used 180 in all the rest of our pipeline as we wanted to maximise the recall. Box plots show the median, the first and third quartile. Top and bottom bars are the maximal and minimal value. Diamonds are outliers.

D-F: Optimisation of the threshold probability volume (x-y-t) used to detect extrusion event (in voxel, xy pixel=0.275μm, t=5 minutes). For each volume, the f1 score (**D**), precision (**E**), and recall (**F**), were estimated on the test dataset. The optimium for the f1score was obtained for 1200, however we used a threshold of 800 for all the rest of the pipeline to maximise the recall. Box plots show the median, the first and third quartile. Top and bottom bars are the maximal and minimal value. Diamonds are outliers.

G-I: Changes in the model to optimise its prediction scores on extrusion. **G:** f1-score. **H:** Recall, **I:** Precision, with the initial two class model (ext), the inclusion of SOPs (ext, SOP), the inclusion of SOPs and cell divisions (ext, SOP, div), including the 3 cellular events and reinforcement (see main text), and using two independent networks (2net). The results shown are the compiling of the prediction of all the testing dataset processed through 4 independent trained networks (except for 2net, which used only one pair of networks). Box plots show the median, the first and third quartile. Top and bottom bars are the maximal and minimal value. Diamonds are outliers.

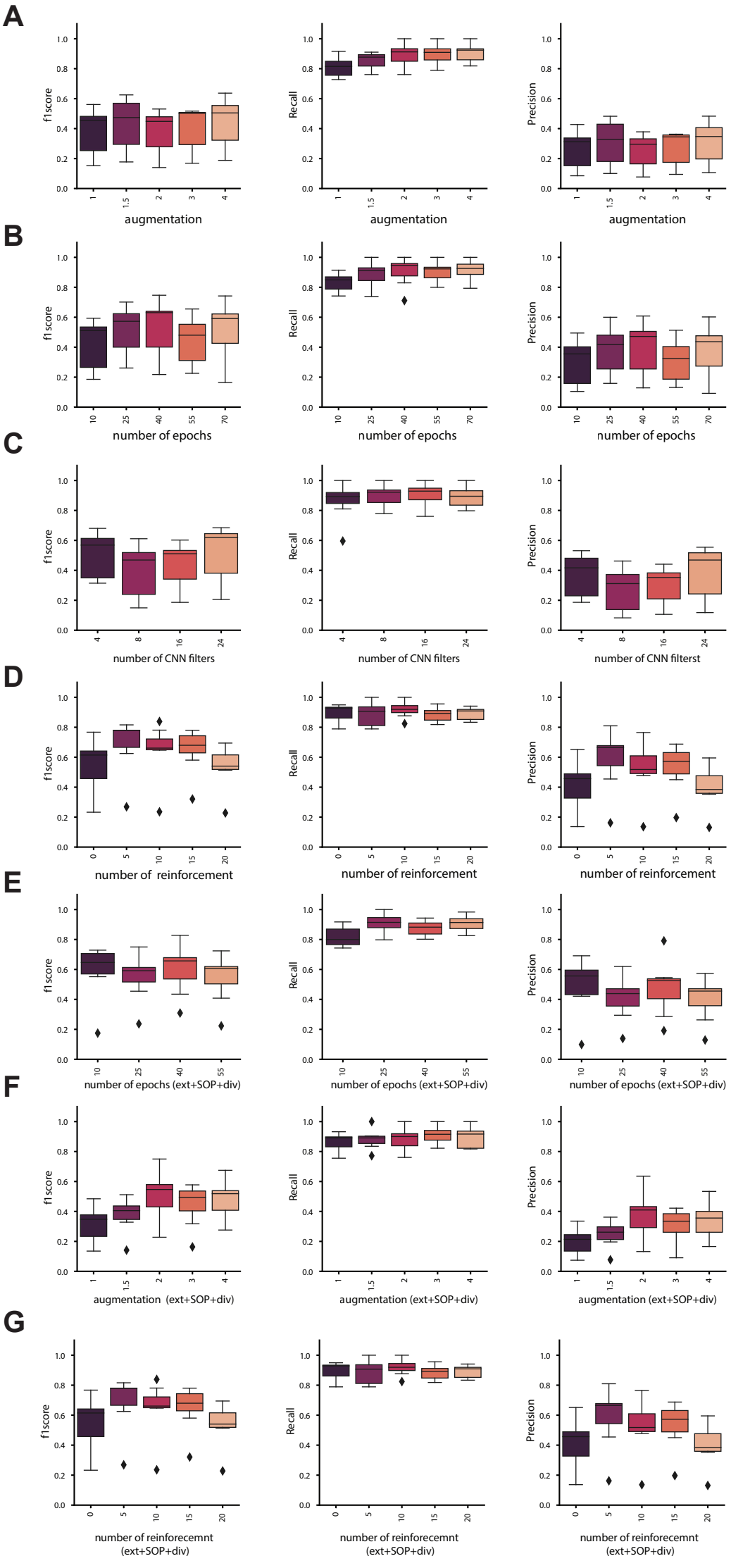

A
f1score
Recall
Precision
augmentation
B
number of epochs
C
number of CNN filters
D
number of reinforcement
E
number of epochs (ext+SOP+div)
F
augmentation (ext+SOP+div)
G
number of reinforcemnt
(ext+SOP+div)

Figure S3: Optimisation of the hyperparameters of DeXNet

A-F: Optimisation of the hyperparameters of DeXtrusion to maximise the precision and the recall. For each set of parameters, the f1 score, Recall and Precision were estimated on the test dataset either for the DeXNet trained only for extrusion detection (**A-D**) or the DeXNet detecting the four categories of events (extrusion, SOPs, division, control) (**E-G**). The parameters used in this study are summarized in the **Supplementary table 3**. These parameters include the number of augmentations of the data (**A, F**), the number of epochs for model fitting (**B, E**), the number of CNN filters (**C**), and the number of reinforcements (**D, G**). Box plots show the median, the first and third quartile. Top and bottom bars are the maximal and minimal value. Diamonds are outliers.

Movie legends

Movie 1: Probability map of extrusions, SOPs and divisions in a WT pupal notum

Local projection of a pupal notum expressing E-cad::GFP (grey) overlayed with the probability map of detection of extrusions (yellow), cell divisions (magenta) and SOPs (cyan). E-cad channel is shown separated on the right. Anterior, left and posterior right. Scale bar=30μm.

Movie 2: Probability map of extrusions, SOPs and divisions in an EGFR depleted pupal notum

Local projection of a pupal notum depleted for EGFR (*pnr-Gal4, UAS-EGFRdsRNA*) expressing E-cad::GFP (grey) overlayed with the probability map of detection of extrusions (yellow), cell divisions (magenta) and SOPs (cyan). E-cad channel is shown separated on the right. Anterior, left and posterior right. Scale bar=30μm.

Movie 3: Probability map of extrusions, SOPs and divisions in a pupal wing

z-projection of a WT pupal wing expressing E-cad::GFP (grey) from (Etournay et al., 2015), overlayed with the probability map of detection of extrusions (yellow) and cell divisions (magenta). E-cad channel is shown separated on the bottom. Distal on the right, proximal on the left. Scale bar=50μm. Note that we only took in consideration the probability overlapping the wing.

Movie 4: Probability map of extrusions in the larval epidermal cells of the pupal abdomen

z-projection of a WT pupal abdomen expressing E-cad::GFP (grey) from (Davis et al., 2022) overlayed with the probability map of detection of extrusions (yellow). E-cad channel is shown separated on the bottom. Scale bar=50μm. Note that we only used the prediction in the LECs and ignored the histoblasts (small cells in the clusters on the left). The cell scale used for this prediction is suboptimal for histoblasts.

Supplementary table 1

Movie	Microscope	Marker	Pixel size (µm)	Frame rate (seconds)	Genotype	extrusion	Cell division	SOP	Nothing	Total # events/movie
1	LSM 880	ubi-Ecad::GFP	0.1037427	60	UAS-hid-RNAi	36	74	60		170
3	spinning disk	ubi-Ecad::GFP	0.275	300	WT	720	237	279		1236
4	spinning disk	ubi-Ecad::GFP	0.275	300	Gal80ts	223	170	196	47	636
5	spinning disk	ubi-Ecad::GFP	0.275	300	Gal80ts	385	62	266	54	767
6	spinning disk	ubi-Ecad::GFP	0.275	300	Gal80ts SPASTIN	563	96	327	26	1012
8	LSM 880	ubi-Ecad::GFP	0.1037427	60	UAS-hid-RNAi	40	3	37	11	91
9	LSM 880	Ecad::GFP(KI)	0.1037427	60	sqh-mCherry	35	40	33		108
10	LSM 880	Ecad::GFP(KI)	0.1037427	60	sqh-mCherry	40	79	68		187
11	LSM 880	Ecad::GFP(KI)	0.1037427	60	sqh-mCherry	18	100	8	26	152
12	LSM 880	Ecad::tdTomato(KI)	0.1037427	20	jupiter-GFP	20	46	40	6	112
13	LSM 880	Ecad::tdTomato(KI)	0.1037427	20	jupiter-GFP	34	117	99	6	256
14	LSM 880	Ecad::tdTomato(KI)	0.1037427	20	SAS4-GFP	59	36	103	9	207
16	LSM 880	Ecad::tdTomato(KI)	0.1037427	300	colcemid	215		45		260
17	LSM 880	Ecad::tdTomato(KI)	0.1037427	300	colcemid	203				203
19	spinning disk	Ecad::tdTomato(KI)	0.275	300	UAS-hid-RNAi ctrl inj	104	123	60	22	309
20	spinning disk	Ecad::tdTomato(KI)	0.275	300	UAS-hid-RNAi ctrl inj	82	116	80	5	283
21	spinning disk	Ecad::tdTomato(KI)	0.275	300	UAS-hid-RNAi ctrl inj	77	487	109	10	683
23	spinning disk	Ecad::GFP(KI)	0.275	300	Gal80ts SPASTIN	566	40	297		903
24	spinning disk	Ecad::GFP(KI)	0.275	300	Gal80ts SPASTIN	464	305	166	25	960
25	LSM 880	Ecad::GFP(KI)	0.18	300	WT	486	331	103		920
27	LSM 880	Ecad::GFP(KI)	0.18	300	WT	878	260	165		1303
28	LSM 880	Ecad::GFP(KI)	0.18	300	EGFR-RNAi	435	60	107	26	628
29	LSM 880	Ecad::GFP(KI)	0.18	300	EGFR-RNAi	489	77	174		740
31	spinning disk	Ecad::GFP(KI)	0.275	300	EGFR-RNAi	528	162	232	31	953
					Total # events/category	6700	3021	3054	304	

Supplementary table 1: Detailed description of the training dataset

Table summarizing the different properties of the microscopy movies used to train our model. These include the number of the movie in our dataset (Movie column), the type of microscope used (Zeiss LSM880 or Gataca Systems Spinning-Disk), the type of E-cadherin marker used to label the cell boundaries (Marker column), the pixel size and the frame rate. Other genetic modifications or transgenes (other markers, RNAis or drug injections) are described in the 'Genotype' column. Finally, this table also displays the number of events labelled for each movie: extrusions, cell divisions, Sensory Organ Precursors cells (SOPs), Number of false positive detected elements further re-labelled as controls in the pipeline optimisation (Nothing) and total number of elements per movie.

Supplementary table 2

Movie	Microscope	Marker	Pixel size (µm)	Frame rate (seconds)	Genotype	Extrusions	Divisions	SOPs	Total # events/movie
2	spinning disk	Ecad::GFP(KI)	0.275	300	WT	883	1351	351	2585
7	spinning disk	ubi-Ecad::GFP	0.275	300	UAS-hid-RNAi	11	26	132	169
15	LSM 880	Ecad::tdTomato(KI)	0.1037427	20	WT	63	89	146	298
18	LSM 880	Ecad::tdTomato(KI)	0.1037427	300	UAS-hid-RNAi colcemid	123			123
22	spinning disk	Ecad::tdTomato(KI)	0.275	300	UAS-hid-RNAi ctrl inj	59	46	193	298
26	LSM 880	Ecad::GFP(KI)	0.18	300	WT	682	120	130	932
30	LSM 880	Ecad::GFP(KI)	0.18	300	EGFR-RNAi	499	108	100	707
					Total events #/category	2320	1740	1052	5112

Supplementary table 2: Detailed description of the test dataset

Table summarizing the different properties of the microscopy movies used to test our model. These include the number of the movie in our dataset (Movie column), the type of microscope used (Zeiss LSM880 or Gataca Systems Spinning-Disk), the type of E-cadherin marker used to label the cell boundaries (Marker column) , the pixel size and the frame rate. Other genetic modifications or transgenes (other markers, RNAis or drug injections) are described in the 'Genotype' column. Finally, this table also displays the number of events labelled for each movie: extrusions, cell divisions, Sensory Organ Precursors cells (SOPs) and total number of elements per movie.

Supplementary table 3

	Hyperparameter	Parameter value	Optimal F1-score
ext. only	Number of augmentation	4	0.504653
	Number of epochs	40	0.630638
	Number of CNN filters	24	0.618855
4 classes	Number of augmentation	2	0.546644
	Number of epochs	40	0.657036
	Number of reinforcement	5	0.779413

Supplementary table 3: Optimisation of model hyperparameters

Table describing the final hyperparameter values used in the final pipeline after careful optimisation and their associated F1-score values. We first optimised the number of augmentation, epochs and CNN filters in the training including extrusions only (ext. only rows). Then we fixed the numbers of CNN filters for the rest of the training. We then included cell divisions and SOPs in order to optimise the pipeline and explored the impact of the number of augmentation and epochs on the F1-score (4 classes rows). Finally, we further optimised the model by adding false positive detected elements further re-labelled as controls in the pipeline optimisation and check the impact on the F1-score (this was done only in the case of 4classes training).

Supplementary table 4

		Generalisation scores	
		Precision	**Recall**
UAS-EGFR-RNAi	Out-of-the-box	0.874	NA
	Retrained	NA	NA
Pupal wing	Out-of-the-box	0.786	0.849
	Retrained	0.909	0.83
LECs	Out-of-the-box	0.738	0.38
	Retrained	0.858	0.69
	Retrained with 2 movies	0.860	0.74

Supplementary table 4: Generalisation scores on unseen movies

Table summarising the generalisation scores (precision and recall) depending on the movie the generalisation was tested on. For each movie checked the scores with the model without further training (out-of-the-box rows) or after retraining (retrained rows). For Larval Epithelial Cells movies (LECs rows) we proceeded to an additional retraining step by including movies from this dataset (retraining with 2 movies).