DeepSea: An efficient deep learning model for automated cell segmentation and tracking

Abolfazl Zargari[1], Gerrald A. Lodewijk[2], Celine W. Neudorf[2], Kimiasadat Araghbidikashani[3], Najmeh Mashhadi[4], Stefany Rubio[5], Lindsay Hinck[5,6], S. Ali Shariati[2,6]

1. Department of Electrical and Computer Engineering, University of California, Santa Cruz, Santa Cruz, CA
2. Department of Biomolecular Engineering, University of California, Santa Cruz, Santa Cruz, CA
3. Department of Chemistry and Biochemistry, University of California, Santa Cruz, Santa Cruz, CA
4. Department of Computer Science and Engineering, University of California, Santa Cruz, Santa Cruz, CA
5. Department of Molecular, Cell and Developmental Biology, Santa Cruz, Santa Cruz, CA
6. Institute for The Biology of Stem Cells, University of California, Santa Cruz, Santa Cruz, CA

* Corresponding author: alish@ucsc.edu

Abstract:

Time-lapse microscopy can directly capture the dynamics and heterogeneity of cellular processes at the single-cell level. Successful application of single-cell live microscopy requires automated segmentation and tracking of hundreds of individual cells over several time points. Recently, deep learning models have ushered in a new era in quantitative analysis of microscopy images. This work presents a versatile and trainable deep-learning-based software, termed DeepSea, that allows for both segmentation and tracking of single cells and their nuclei in sequences of phase-contrast live microscopy images. We show that DeepSea can quantify several cell biological features of mouse embryonic stem cells, such as cell division cycle, mitosis, cell morphology, and cell size, with high precision using phase-contrast images. Using DeepSea, we were able to show that despite their ultrafast cell division cycle, mouse embryonic stem cells exhibit cell size control in the G1 phase of the cell cycle.

Introduction:

Cells are frequently adapting their behavior in response to environmental cues to make important fate decisions, such as whether to divide or not. In addition, individual cells within a clonal population and under identical culture conditions show heterogeneity in response to environmental cues [1]. In recent years, it has become clear that single-cell level analysis over time is essential for revealing the dynamics and heterogeneity of individual cells [2, 3].

Single-cell quantitative live microscopy can directly capture both dynamics and heterogeneity of cellular decisions by continuous long-term measurements of cellular features [4, 5]. Widely available microscopy techniques such as label-free phase-contrast live microscopy allow for monitoring the dynamics of morphological features such as the size and shape of the cells. The key to the successful application of single-cell live microscopy is the scalable and automated analysis of a large dataset of images. Typical live-cell imaging of biological features of cells is a

multi-day experiment that produces several gigabytes of images collected from thousands of cells. A major challenge for quantitative analysis of these images is the difficulty of accurately segmenting individual cells and their subcellular organelles and tracking them over time. Low signal-to-noise ratio, the close proximity of cells, and unpredictable movements are among the challenges for software-based automated image analysis of live single-cell microscopy data. In addition, cells are non-rigid bodies, and thus, tracking them is more challenging because they can change their shapes with time. Most critically, they divide into two new daughter cells during mitosis, which is unique and not comparable with other phenomena we encounter in conventional object tracking applications. Solving single-cell microscopy challenges requires integrating different disciplines, such as cell biology, image processing, and machine learning.

In recent years, deep learning (DL) has outperformed conventional rule-based image processing techniques in tasks such as object segmentation and object tracking [6-8]. Traditional image segmentation approaches often require experiment-specific parameter tuning, while DL schemes are adaptive and trainable. More recently, deep learning-based image processing methods have attracted attention among cell biologists and microscopists, for example, to localize single-molecule in super-resolution microscopy [9], to enhance the resolution of fluorescence microscopy images [10], to develop an automated neurite segmentation system using a large 3D anisotropic electron microscopy image dataset [11], to design a model to restore a wide range of fluorescence microscopy data [12], and to train a fast model that refocuses 2D fluorescence images onto 3D surfaces within the sample [13].

Here, we developed a versatile and trainable deep learning model for cell body segmentation and cell tracking in phase-contrast microscopy images of mouse embryonic stem cells (mESCs) at the single-cell level. Using this model, we developed a user-friendly software, termed DeepSea, for automated and quantitative analysis of phase-contrast microscopy images. We showed that DeepSea is able to capture dynamics and heterogeneity of cellular features such as cell cycle division and cell size in mESCs. Our analysis of cell size in mESCs revealed that mouse embryonic stem cells exhibit cell size control in the G1 phase of the cell cycle.

Results:

Training a cell segmentation and tracking model:

An overview of our approach is presented in Figure1. To generate a dataset of stem cell image sequences, we performed time-lapse imaging of mESCs for up to 48 hours. We used 31 sets of time-lapse phase-contrast microscopy image sequences to build our dataset (Figure 1A). Individual mESCs were manually annotated and labeled to generate a training dataset, including pairs of original mESCs images and corresponding cell ground-truth mask images, using our in-house Matlab-based cell segmentation software (https://deepseas.org/software/). Next, we used the annotated dataset of stem cell images to train a supervised deep learning algorithm. We developed and trained an automated model called DeepSea to detect and segment the cell and nucleus bodies (Figure 1B). Our model is composed of 13 layers and five residual blocks (Figure 1B, Figure 6).

We trained two independent cell segmentation models: one for the cell area and a second for the nucleus area segmentation. By combining the output of these two models and applying a watershed technique, we were able to automate cell and nucleus segmentation by developing an efficient and reliable deep learning-based segmentation model that separates the cell and nucleus regions from other parts of the cell images.

To be able to visualize the dynamics of cellular behavior, we added cell tracking capability to our DeepSea model (Figure 1C). We trained a DL model with a structure similar to the DeepSea detection model involving two separate input paths fed by cell images of times t-τ and t (segmented by DeepSea segmentation model). With this model, we were able to monitor multiple cellular phenotypes and several cell division cycles across the microscopy image sequences and generate lineage trees of cells (Figure 1D, Figure 2).

To make our model widely accessible, we developed a deep learning-based software with a graphical user interface that allows researchers with no background in machine learning to automate the measurement of cellular and subcellular features of live microscopy data (Figure 1D). Some editing options embedded in this software help users correct our model outputs if needed and obtain more precise results. Our user-friendly software and dataset are publicly available on https://deepseas.org/.

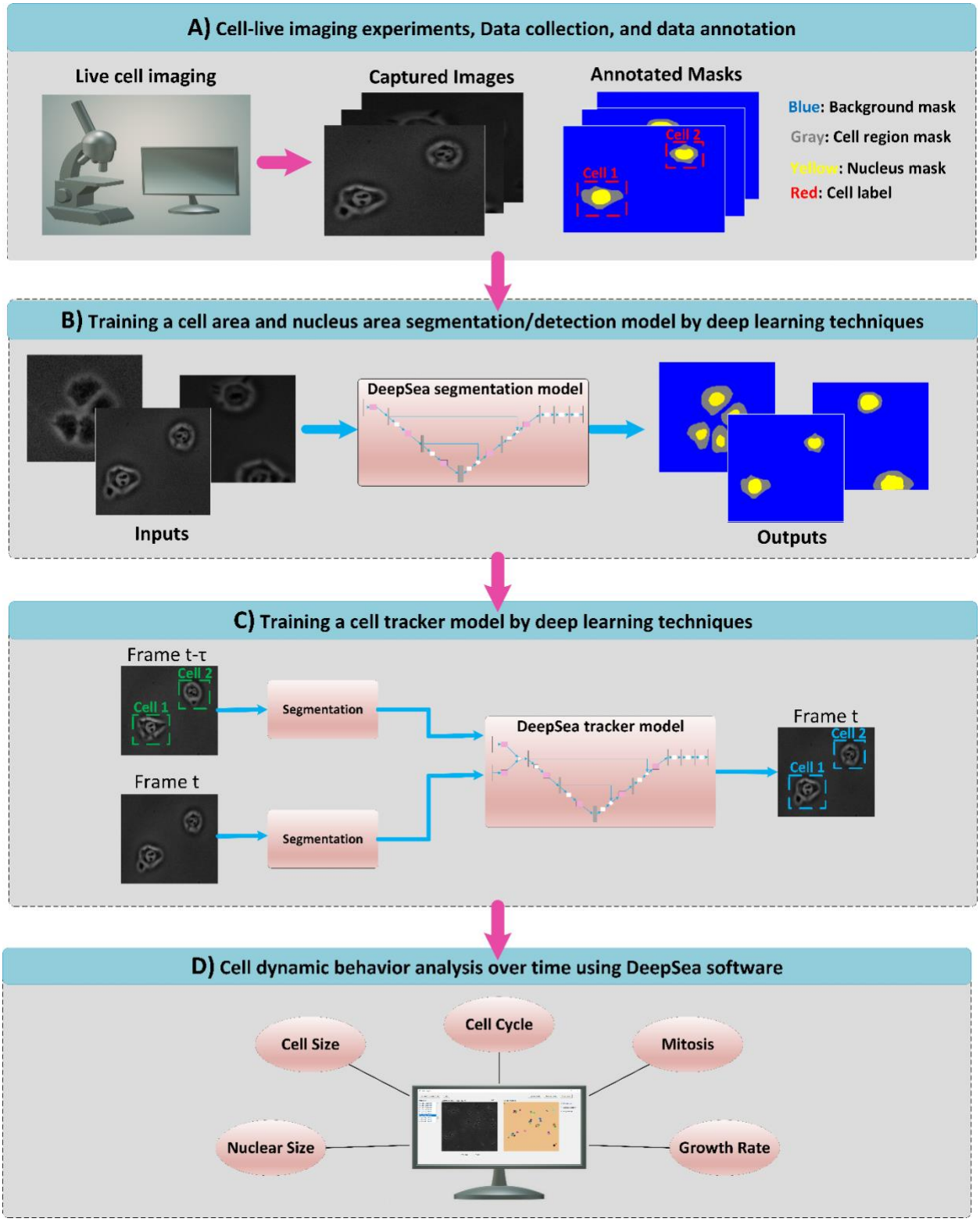

Figure 1: An overview of our DeepSea software development and its features

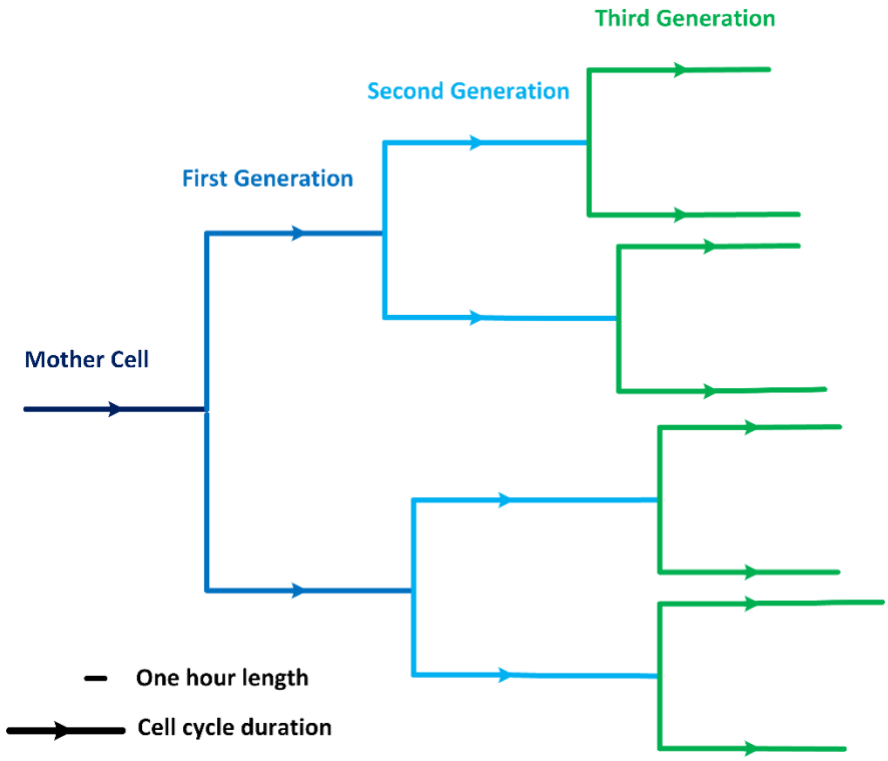

Figure 2: Lineage trees created by automated tracking a single cell over three generations

DeepSea performance evaluation

We evaluated our trained models on the DeepSea test images. Our results were obtained from three evaluations of segmentation, detection, and tracking performances. In the segmentation process, the trained model fits the exact boundary of the target cells and labels their pixels with different colors, helping to determine each cell's shape and area within the input microscopy image. In the detection process, we classify our segmented objects into the cell or non-cell categories using Intersection Over Union (IOU) as the detection metric. Also, in the tracking process, the tracker model uses the feature map and location of the segmented target cells within the previous frame to localize and track them with the same label among the segmented cells in the current frame.

Figures 3A and 3B show our trained model's performance using common metrics used in pixel-wise segmentation and object detection tasks [14, 15]. Results showed that all the cells' evaluation scores are higher than the nuclei scores, mainly due to the low contrast texture of cells' internal regions on phase-contrast images. In addition, our model's predictions upon encountering moderate and hard cases are still as accurate as easy cases in cell area segmentation and detection tests. Two examples of the DeepSea segmentation model's output are also shown in Figure 3C.

For the tracker model, we first evaluated our trained model in a frame-by-frame tracking test. We used our test set images, including pairs of cropped search windows at times t-τ (previous frame)

and t (current frame). In addition to the overall assessment, we analyzed our model by applying the easy case, hard case, single-cell, and birth tracking challenges (Figure 4A). Our model achieved a precision of more than 93% in all scenarios and showed almost the same results in single-cell tracking and birth tracking tests (precision=94% and 93%, respectively). In another evaluation approach, we assessed our tracker model in a full cell cycle tracking task. This test uses the trained model to track all the cell motion trajectories across the live-cell microscopy frame sequences. Figure 4B shows two examples of our model output, tracking the cell motion trajectories over three consecutive frames. We fed our model with four video sets (including 137 full cell cycle trajectories) and then measured different scores (usually used in conventional multi-object tracking schemes [16, 17]), listed in Table 1. To be able to test the accuracy of our tracking model, we used MOTA (Multi-Object Tracking Accuracy) that measures the precision of localizing objects over time. Our tracker model achieved a MOTA value of 86% with 79% of cell cycle trajectories are nearly completely tracked.

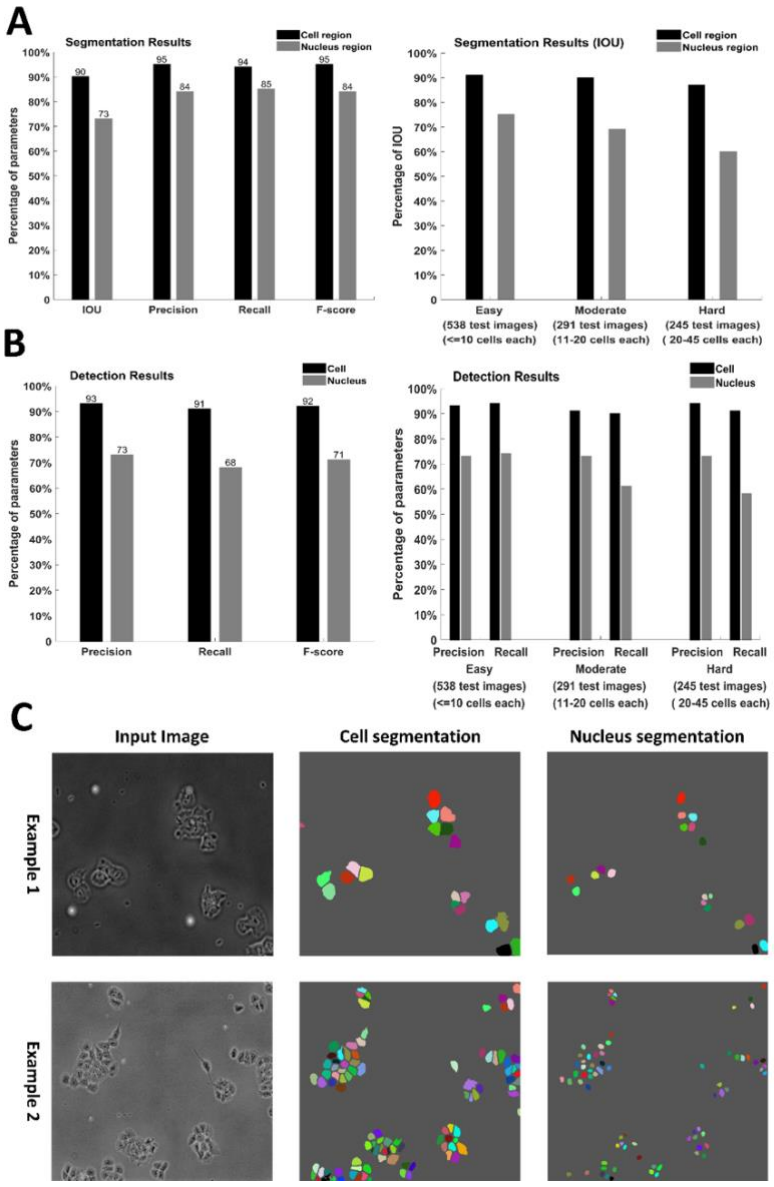

Figure 3: Segmentation model evaluation for the test set. A) We evaluated the Pixel-wise segmentation performance and the trained model using mean IOU, precision, recall, F-score measurement. We also used the IOU metric as an index to assess our model dealing with test cases with different difficulty levels. B) We used IOU measurement as a detection index, relying on the assumption that an IOU measurement greater than 0.7 means successful detection of the cell/nucleus body. Then, we calculated different conventional metrics by recognizing true-positive, false-positive, and false-negative detections to evaluate our model detection capability. C) Two examples of our model outputs for the cells and nuclei body segmentation. Each color represents an individual cell or nucleus region.

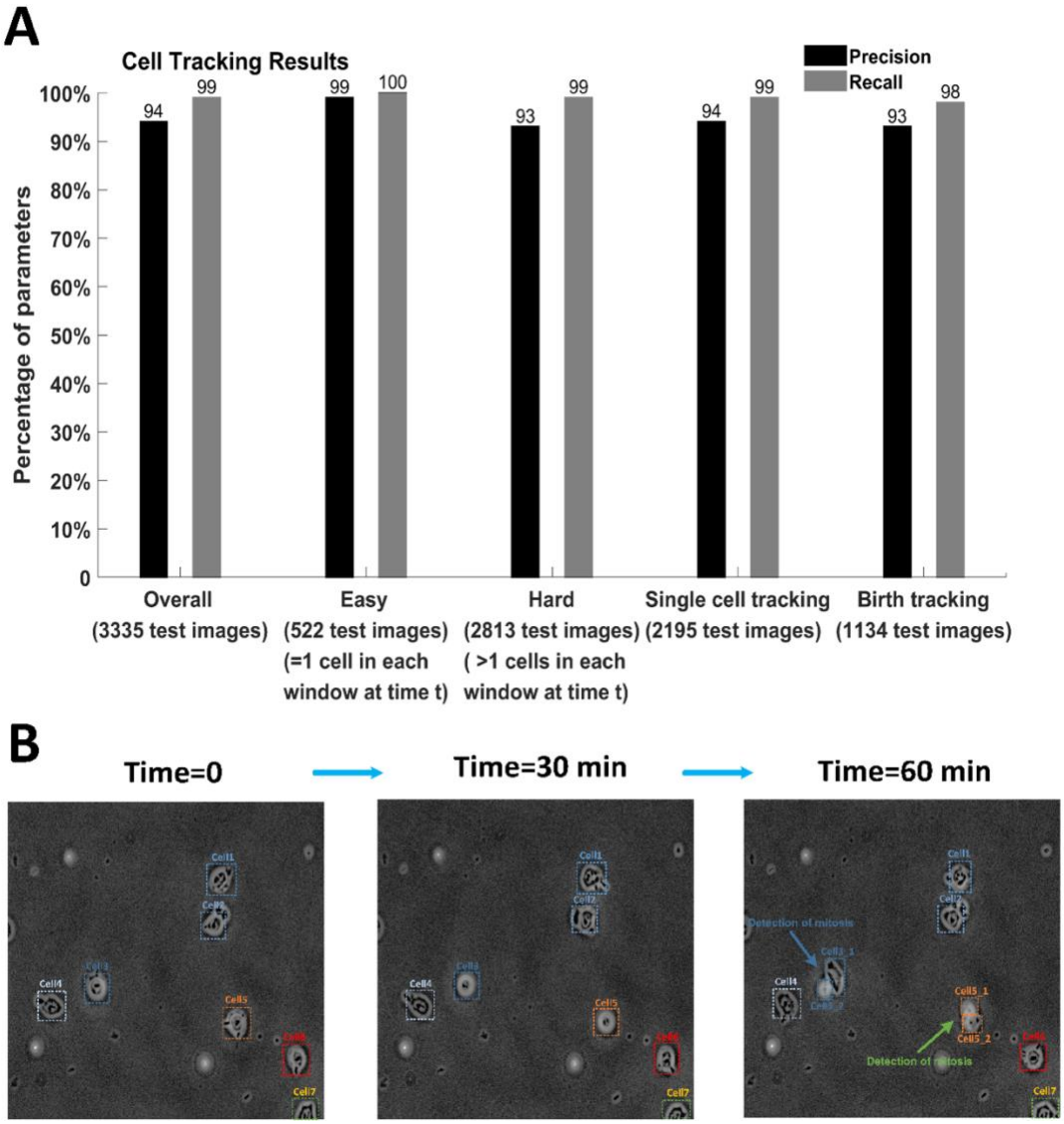

Figure 4: The tracker model's performance. A) We evaluated our trained tracker model in the window-by-window cell tracking task using the test set and considering precision and recall computations. We tested our model given the overall performance and difficulty levels of easy, hard, single-cell tracking, and birth tracking. B) Example of the cell cycle tracking process obtained by feeding three consecutive cell frames (with the time difference of 30 minutes) to our trained tracker model. Daughter cells are linked to their mother cells by underline (in the third frame).

Table 1: Multi-cell cycle tracking results

MOTA	MT	ML	Precision	Recall	Frag	IDS	Number of cell cycles	Number of detected cells
0.86	0.79	0.11	0.95	0.91	49	27	137	2576

Cell size regulation of mouse embryonic stem cells

Cells need to grow in size before they can undergo division. Different cell types maintain a fairly uniform size distribution by actively controlling their size in the G1 phase of the cell cycle [18]. However, the typical G1 control mechanisms of somatic cells are altered in mESCs [19]. Mouse embryonic stem cells have an unusually rapid cell division cycle that takes about 10h to be completed (Figure 5A). The rapid cell cycle of mESCs is primarily due to an ultrafast G1 phase that is about 2h compared to ~10 in skin fibroblast cells (Supplementary Figure 2B). An interesting question is whether mESCs can employ size control in their rapid G1 phase, just as most somatic cells do.

To answer this question, we used our trained DeepSea model to analyze cell size control in mESCs. Using confocal microscopy, we showed that the area of a cell is closely correlated with the cell volume, making the area a faithful measurement for cell size (Supplementary Figure 1). By measuring the size of the sister cells at birth, we showed 42% of divisions results in daughter cells of different sizes (Figure 5B). We hypothesized that smaller born cells would spend more time growing the cell cycle compared to their larger sister cells. In support of this hypothesis, we observed that smaller born cells increase their cell cycle duration by about ~2h compared to their larger sister cells (Figure 5C). Next, we asked if the variations in cell size at birth will be corrected over the course of one cell division cycle by measuring the coefficient of variation (CV) of cell size throughout the cell cycle (Figure 5D). We observed a rapid decline in size variation within the first 3h after birth supporting the hypothesis that mESCs regulate their size in the G1 phase of the cell cycle. Together, our results support the hypothesis that the mESCs can adjust the cell cycle based on birth size, suggesting cell size control through an unknown molecular mechanism [18].

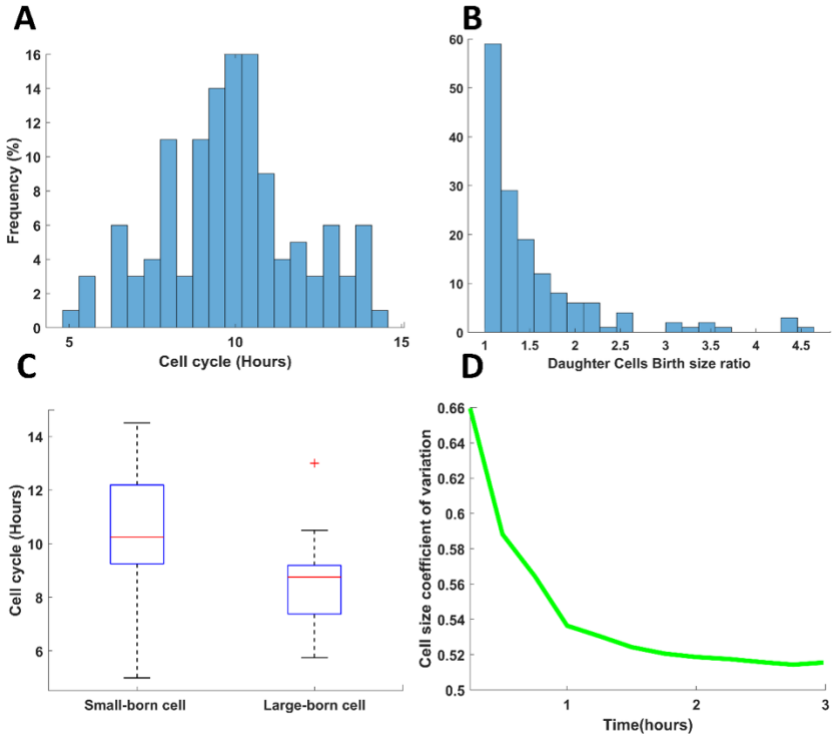

Figure 5: Cell size regulation in mouse embryonic stem cells. A) Distribution of the cell cycles. B) Histogram of birth size ratio of daughter cell pairs. C) Comparing the cell cycle duration of the cells that were born small (smaller than a third of the measured cell area range) with those born large (larger than two-third of the measured cell area range). D) Cell size coefficient of variation.

Discussion:

Here we introduced DeepSea, a deep learning model for automated analysis of time-lapse images of cells and their nuclei. The segmentation and tracking of cell bodies and subcellular organelles from microscopy images are critical steps for nearly all microscopy-based biological analysis applications. Although phase-contrast microscopy is a non-invasive and widely-used method for live-cell imaging, developing automated segmentation and tracking algorithms remains challenging. This is due to low-quality imaging systems, the unpredictable nature of cells in their movements over time, and the close proximity of cells. Here, we leveraged the recent advancement in deep learning-based image processing to address some of these challenges.

The lack of a comprehensive annotated dataset prevents the full utilization of deep learning-based models for microscopy image analysis systems. We generated large manually annotated datasets of time-lapse microscopy images of mESCs, which are publicly available and can be used for new image analysis models. In addition, we were able to significantly increase the size of annotated data covering more variations by applying image augmentation techniques, which benefited from both conventional image augmentation techniques and a proposed random cell movements method. We expect this resource to facilitate the future application of deep learning-based models for the analysis of microscopy images.

To address the challenge of cell and nucleus segmentation and tracking, we built a deep learning model, termed DeepSea, which can efficiently segment cell and nucleus areas in phase-contrast microscopy images. Our segmentation model was trained on our generated dataset, containing over 100000 segmented cell and nucleus regions, and achieved an IOU value of 90% and an F-score value of 95% in the cell body segmentation task. Most importantly, we were able to exploit the deep learning capabilities to automate the tracking of cells across the time-lapse microscopy image sequences. Due to the inherent complexity of the dataset, we introduced an efficient scheme and prepared new tracking data (containing over 16000 cases) to be used in the test and training process of the DeepSea cell tracker. Our DeepSea tracker model was able to track the full cell cycle trajectories with a MOTA value of more than 86% obtained from 2576 detected cells and 137 cell cycles.

We showcased the application of the DeepSea by investigating cell size regulation in mESCs across hundreds of cell division cycles. Our cell size analysis revealed that smaller born mESCs regulate their size by spending more time growing in the G1 phase of the cell cycle. These findings strongly support the idea that mESCs actively monitor their size consistent with the presence of size control mechanisms in the short G1 phase of the embryonic cell cycle.

We would like to note that our dataset and models are limited to the phase contrast 2D images of mESCs. A larger dataset of the samples from different cell types and different imaging modalities would be useful for testing our proposed model's generalization, reliability, and robustness. In addition, in our future work, we will investigate other deep models that have recently achieved considerable advancement in object detection and tracking tasks, such as Faster-RCNN,

Recurrent Yolo, TrackR-CNN, JDE, RetinaNet, CenterPoint, and so forth [6- 8], or merge their architecture with our current models to improve the results.

Method:

Cell Culture and Microscopy:

Mouse ESCs (V6.5) were maintained on 0.1% gelatin-coated cell culture dishes in 2i media (Millipore Sigma, SF016-100) supplemented with 100U/ml Penicillin-Streptomycin (Thermo Fisher, 15140122). Cells were passaged every 3-4 days using Accutase (Innovate Cell Technologies, AT104) and seeded at a density of 5,000-10,000 cells/cm2. For live imaging, between 5000 to 10,000 cells were seeded on 35mm dishes with a laminin-coated (Biolamina) 14mm glass microwell (MatTek, P35G-1.5-14-C). Cells were imaged in a chamber at 37C perfused with 5% CO2, a Zeiss AxioVert 200M microscope with an automated stage, and an EC Plan-Neofluar 5x/0.16NA Ph1objective or an A-plan 10x/0.25NA Ph1 objective. The same culture condition was used for confocal imaging except that 24 hours after seeding, media was replaced with 2ml DMEM-F12 (Thermo Fisher, 11039047) containing 2ul CellTracker Green CMFDA dye (Thermo Fisher, C2925) and placed back in the incubator for 35 minutes. Next, 2 ul of CellMask Orange plasma membrane stain (Thermo Fisher, C10045) was added, and the dish was incubated for another 10 minutes. Dishes were washed three times with DMEM-F12, after which 2ml of fresh 2i media was added. Cells were imaged directly after the live-cell staining protocol using the Zeiss 880 Microscope using a 20x/0.4 N.A. objective and a 1µm interval through the z-axis.

Data Collection:
Supplementary Figure 3 shows the block diagram of preparing our dataset. We collected 31 sets of phase-contrast time-lapse microscopy image sequences, including 1074 frames initially (Supplementary Figure 3A), with the characteristics listed in Table 2. Our collected datasets are publicly available on https://deepseas.org/datasets/. In the next step, we improved image contrasts using an adaptive histogram equalization (AHE) algorithm (Supplementary Figure 3B) [20]. AHE is able to enhance subtle changes in the noisy image regions and effectively used to improve low-contrast images with uneven illuminations. We designed a cell segmentation software in MATLAB (Supplementary Figure 3C and https://deepseas.org/software/) to manually create the ground-truth mask image corresponding to each stem cell image (Supplementary Figure 3D). We applied a modified data augmentation scheme to generate a larger dataset efficiently and less expensively (Supplementary Figure 3E). The image augmentation technique offers a solution to create new annotated samples by modifying existing data in a controlled way using a set of image-based or feature-based transformations. Some of these transformations, usually used in image augmentation tasks, are rotation, flipping, reflection, wrapping, adding noise, color space shifting, scaling, cropping, padding, and brightness shifting [21]. In our image augmentation scheme, in addition to some of the conventional image transformations, we proposed to move the stem cell bodies by the random vectors of (θ,d), where θ is the direction angle between 0 and 360 and d is displacement in pixels. An example of the input and output of the image augmentation block is shown in Supplementary Figure 4. Using the proposed method, we could generate a dataset of 5370 annotated stem cell images (Supplementary Figure 3F).

Table 2: Original collected raw image dataset characteristics

Characteristic	
Frame rate	20 min (7 sets and 284 frames) or 30 min (22 sets and 731 frames) or 15 min (2 sets and 59 frames)
Width	1344
Height	1024
Resolution	1.5504 pixels per micron (7 sets and 284 frames) or 3.1008 pixels per micron (24 sets and 790 frames)
Pixel size	0.6450x0.6450 micron^2 (7 sets and 284 frames) or 0.3225x0.3225 micron^2 (24 sets and 790 frames)
Bits per pixel	16 (unsigned)

For the cell body tracking process, we needed to prepare a different dataset that was specially created and structured for our proposed tracker model. For this purpose, we first extracted image triples of the annotated dataset we created for the segmentation task (Supplementary Figure 5A). Then, as shown in Supplementary Figure 5B, for each target cell in the video frame captured at time t-τ, we cropped a rectangular window with a margin of 15 pixels away from the target cell bounding box in all directions. At the same time, we remove other non-target segmented cells. Also, we used the same window coordinates to crop the corresponding rectangular area on the cell image and the mask image at time t. In the tracking process, the goal is matching the target segmented cell body at time t-τ to one of the cells segmented at time t. The matching process relies on one key assumption that the target cell moves slightly during the time distance of τ, and thus, most likely, it or the main part of it can be tracked and detected in the same search rectangular window as the selected window at time t-τ. Accordingly, in the final tracking dataset, we created 16511 triples of cropped target cell windows at time t-τ, cropped corresponding windows at time t, and cropped target cell mask at time t (Supplementary Figure 5C).

Segmentation model

We proposed and built a deep learning model based on the scaled-down version of U-Net architecture and called it DeepSea to be used in the segmentation task (Figure 6). Our model employs residual blocks aiming to increase the model's depth and improve model performance, as observed in Residual Networks [22]. Our DeepSea model includes ~2,000,000 parameters indicating that it is considerably smaller than typical object segmentation models such as UNET [23], PSPNET [24], SEGNET [25], and so forth. We also took advantage of batch normalization and dropout techniques to improve the model's speed, performance, and stability. Supplementary Figure 6 depicts the process we performed to train and test our DeepSea model in the segmentation process. We used 70% of the augmented dataset for the training process, and thus 30% of the remaining data are used in the validation and test process to evaluate the trained model's performance. To further improve our models' outputs, we took benefits from the

watershed technique [26]. As a complementary technique, the watershed helped us separate the remaining segmented cell bodies, which are still touched in some small boundaries.

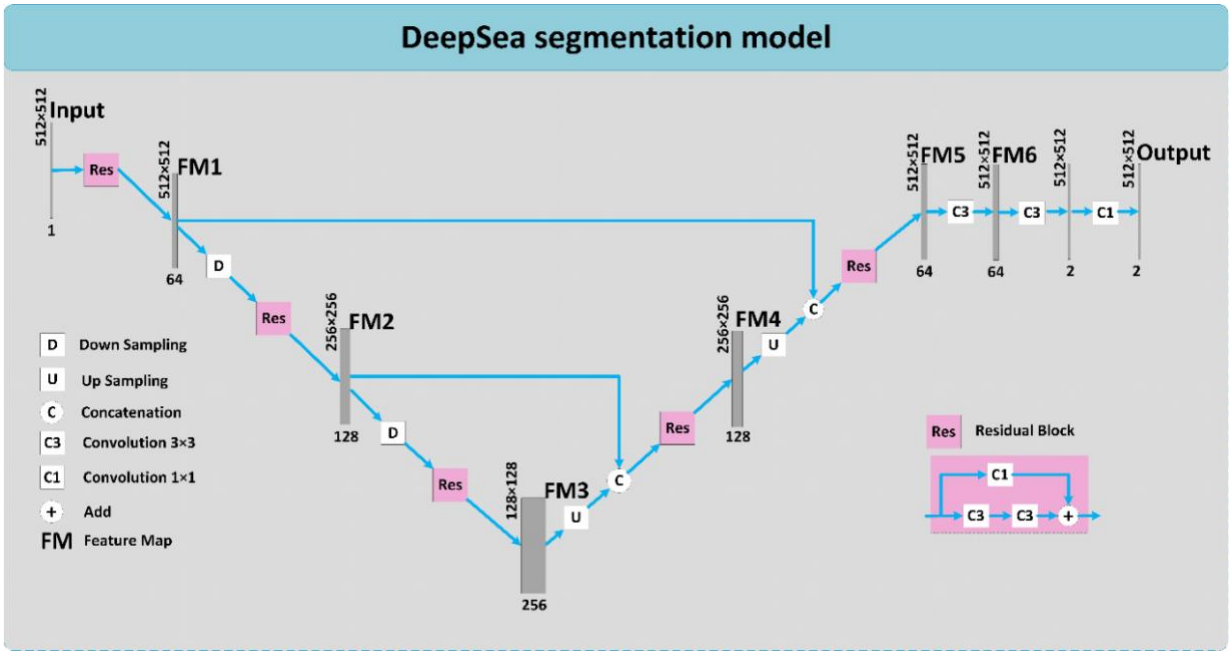

Figure 6: DeepSea segmentation model architecture

Tracking model

For the cell tracking task, we used the same architecture as we employed in the segmentation task with two inputs at time t and time t-τ (Figure 7). To localize the target cell among the segmented cells in the current frame (at time t), the tracker model should know where the target cell was previously located (at time t-τ). Since cells move slowly through space, the cell's previous location will present a good guess of where the model should expect to find it in the current frame. As mentioned in the data collection section, we achieve this by choosing a rectangular search window in our current frame based on the previous bounding box location of target cell (at time t-τ). We crop the previous and current frames by the same search window and input these crops into our convolutional tracker model (Figure 7). The goal of the tracker model is then to localize and segment the current location of the target cell within the search window at time t. The training and test processes are shown in Supplementary Figure 7. We applied 60% of the prepared tracking dataset in the training phase, and the rest were used to evaluate the performance of the trained model in the validation and test phases.

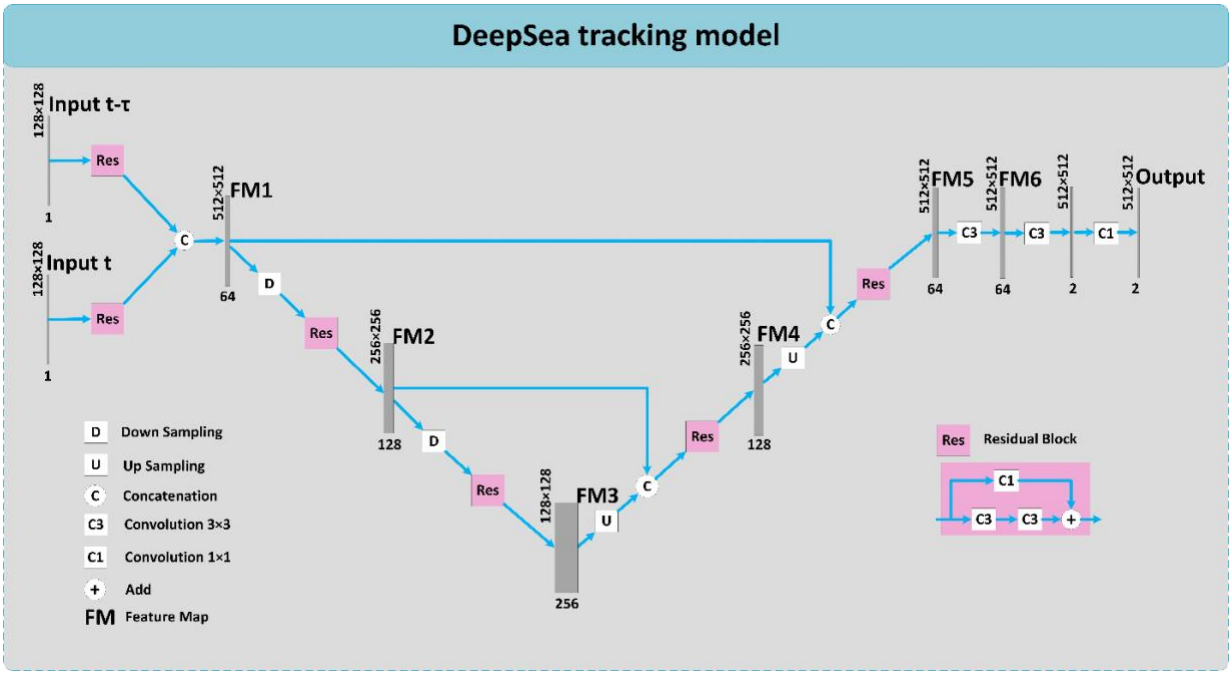

Figure 7: DeepSea tracking model architecture

Models training hyperparameters

For the training process of the proposed DeepSea models, we chose Adam optimizer to optimize model weights and minimize the proposed loss function (Equation 1). Our loss function is a linear combination of cross-entropy (CE) loss and Dice loss (DL) functions, popularly used in image segmentation tasks [27]. CE takes care of pixel-wise prediction accuracy, while DL leads the learning algorithm to increase the overlap between true area and predicted area, which is essentially needed where the number of image background pixels is much higher than foreground pixels (object area pixels).

$$Loss = CE + DL \qquad (1)$$

Table 3 shows the learning algorithm hyperparameters set during the training processes.

Table 3: The learning algorithm hyperparameters

	Segmentation and Detection process	Tracking process
MaxEpochs	100	100
BatchSize	4	16
LearningRate	0.0001	0.0001
ValidationRatio	0.15	0.2
TestRatio	0.15	0.2
TrainRatio	0.7	0.6
DropOut	0.2	0.2

Evaluation Metrics

The most important evaluation metric we used in the segmentation analysis is Intersection Over Union (IOU), known as the Jaccard index and defined as follows [28]:

$$\text{IOU} = \frac{Area\ of\ overlap\ between\ predicted\ pixels\ and\ ground\ truth\ pixels}{Area\ of\ union\ encompassed\ by\ both\ predicted\ pixels\ and\ ground\ truth\ pixels} \quad (2)$$

Other metrics we used in the test and evaluation process are listed in the following Equations (3-5) [14, 15]:

$$\text{Precision} = \frac{TP}{TP + FP} \quad (3)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (4)$$

$$\text{F} - \text{Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (5)$$

Where TP is the true positive, FP is the false positive, and FN is the false-negative predictions. We would like to note that we used the IOU value as an index to evaluate our model's object detection power. We hypothesized that if the IOU value (measured for each segmented cell region) is more than 0.7, the cell is assigned successfully detected from non-cell objects. To evaluate the performance of the tracker model, we measured MOTA (Multiple Object Tracking Accuracy) specially used multi-object tracking challenges along with other metrics as follows [16, 17].

IDS: Identity switches is the number of times a cell is assigned a new label in its track.
MT: Mostly tracked targets is the number of target cells assigned the same label for at least 80% of the video frames.
ML: Mostly lost targets is the number of target cells assigned the same label for at most 20% of the video frames.
Frag: Fragmentation is the number of times a cell is lost in a frame but then redetected in a future frame (fragmenting the track).

$$MOTA = \frac{\sum_n (FP_n + FN_n + IDS)}{\sum_n (Number\ of\ cells)} \quad (6)$$

, Where n is the frame number. A perfect tracker model achieves MOTA=1.

Software availability

All implemented methods are provided as the Python scripts and can be downloaded from https://github.com/abzargar/DeepSea. Also, our DeepSea software is available on https://deepseas.org/software/ with tutorials. DeepSea is a user-friendly software designed to enable researchers to 1) load and explore their phase-contrast cell images in a high contrast

display, 2) detect and localize cell and nucleus bodies, 3) extract some useful features from segmented cell regions, and 4) track and label cell lineages across the frame sequences. It employs our last trained DeepSea models in the segmentation and tracking processes. Users can also manually edit the software productions using the edit options and then store them in their local systems.

Author Contribution: A.Z.K and S.A.S conceived the project and wrote the manuscript. S.A.S performed live microscopy imaging. A.Z.K developed the DeepSea software and implemented the machine learning code. A.Z.K, N.M, C.W.N and K.A contributed to the generation of a manually annotated dataset of images. G.L, S.R and L.H contributed to the confocal microscopy images and provided feedback to improve our manuscript.

Acknowledgment:

This work was supported by the NIGMS/NIH through a Pathway to Independence Award K99GM126027 (S.A.S.) and start-up package of the University of California, Santa Cruz.

References

[1] J. Fiorentino, M. Torres, and A. Scialdone, "Measuring and Modeling Single-Cell Heterogeneity and Fate Decision in Mouse Embryos," Annual Review of Genetics, Vol. 54, 2020.
[2] Bogdan P. et al., "Heterogeneous Structure of Stem Cells Dynamics: Statistical Models and Quantitative Predictions," Nature Scientific Reports, Vol. 4, 2014.
[3] Semrau S. et al.," Dynamics of lineage commitment revealed by single-cell transcriptomics of differentiating embryonic stem cells," Nature Communications, Vol. 8, 2017.
[4] Stavroula Skylaki, Oliver Hilsenbeck & Timm Schroeder, "Challenges in long-term imaging and quantification of single-cell dynamics," Nature Biotechnology, Vol. 34, 2016.
[5] Anatole Chessel; Rafael E. Carazo Salas, "From observing to predicting single-cell structure and function with high-throughput/high-content microscopy," Vol. 63, No. 2, pp. 197–208, 2019.
[6] G. Ciaparrone et al., "Deep learning in video multi-object tracking: A survey," Neurocomputing, Vol. 381, pp. 61-88, 2020.
[7] S. Yun and S. Kim, "Recurrent YOLO and LSTM-based IR single pedestrian tracking," IEEE Conference on Control, Automation and Systems, pp. 15-18, 2019.
[8] X. Zhou, D. Wang, P. Krähenbühl, “Objects as Points,” arXiv:1904.07850, 2019.
[10] P. Voigtlaender et al., "MOTS: Multi-Object Tracking and Segmentation," IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2019.
[9] Ouyang, W. et al. Deep learning massively accelerates super-resolution localization microscopy. Nat. Biotechnol. 36, 460–468 (2018).
[10] Wang, H. et al. Deep learning enables cross-modality super-resolution in fluorescence microscopy. Nat. Methods 16, 103– (2019).
[11] Beier, T. et al. Multicut brings automated neurite segmentation closer to human performance. Nat. Methods 14, 101–102 (2017).
[12] Weigert, M. et al. Content-aware image restoration: pushing the limits of fluorescence microscopy. Nat. Methods, 15, 1090–1097 (2018).

[13] Wu, Y. et al. Three-dimensional virtual refocusing of fluorescence microscopy images using deep learning. Nat. Methods, 16, 1323–1331 (2019).
[14] Liu, L. et al., "Deep Learning for Generic Object Detection: A Survey," International Journal of Computer Vision, Vol. 128, pp. 261–318, 2020.
[15] Minaee, S. et al., "Image Segmentation Using Deep Learning: A Survey," arXiv:2001.05566v5, 2020.
[16] Ciaparrone, G. et al., "Deep Learning in Video Multi-Object Tracking: A Survey," Computer Vision and Pattern Recognition, arXiv:1907.12740, 2019.
[17] Ristani, E. et al., "Performance Measures and a Data Set for Multi-Target, Multi-Camera Tracking," Computer Vision and Pattern Recognition, arXiv:1609.01775, 2016.
[18] Evgeny Zatulovskiy and Jan M. Skotheim, "On the Molecular Mechanisms Regulating Animal Cell Size Homeostasis," Trends Genetics, Vol. 36, pp. 360–372, 2020.
[19] Ben Boward, Tianming Wu, and Stephen Dalton, "Concise Review: Control of Cell Fate Through Cell Cycle and Pluripotency Networks," Stem Cells, Vol. 34, pp. 1427-1436, 2016.
[20] Yadav, G. et al., "Contrast limited adaptive histogram equalization based enhancement for real time video system," International Conference on Advances in Computing, Communications and Informatics (ICACCI), pp. 24-27, 2014.
[21] Connor Shorten & Taghi M. Khoshgoftaar, "A survey on Image Data Augmentation for Deep Learning," Journal of Big Data, Vol. 6, No. 60, 2019.
[22] Khan, A. et al., "A survey of the recent architectures of deep convolutional neural networks," Artificial Intelligence Review, Vol. 53, pp. 5455–5516, 2020.
[23] Olaf Ronneberger, Philipp Fischer, Thomas Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," Computer Vision and Pattern Recognition, arXiv:1505.04597, 2015.
[24] Zhao, H. et al., "Pyramid Scene Parsing Network," Computer Vision and Pattern Recognition, arXiv:1612.01105, 2017.
[25] Vijay Badrinarayanan, Alex Kendall, Roberto Cipolla, "SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation," Computer Vision and Pattern Recognition, arXiv:1511.00561, 2016.
[26] Jos B.T.M. Roerdink and Arnold Meijster, "The Watershed Transform: Definitions, Algorithms and Parallelization Strategies," Fundamenta Informaticae, vol. 41, no. 1,2, pp. 187-228, 2000.
[27] Shruti Jadon, "A survey of loss functions for semantic segmentation," Image and Video Processing, arXiv:2006.14822, 2020.
[28] Rezatofighi, H. et al., "Generalized Intersection over Union: A Metric and A Loss for Bounding Box Regression." Computer Vision and Pattern Recognition, arXiv:1902.09630, 2019.